Software Quality Journal
https://link.springer.com/article/10.1007%2Fs11219-019-09462-5

The skills that employers look for in software testers

Raluca Florea & Viktoria Stray
Department of Informatics, University of Oslo, Norway

Cite this article

Florea, R., Stray, V. The skills that employers look for in software testers. *Software Qual J* **27,** 1449–1479 (2019). https://doi.org/10.1007/s11219-019-09462-5

The Skills That Employers Look for in Software Testers

Raluca Florea ✉ [1], Viktoria Stray[1]

[1] University of Oslo, Gaustadalléen 23 B, 0373 Oslo, Norway

ralucamf, stray@ifi.uio.no

Abstract
Software testing is an integral part of software development that provides better quality products and user experiences and helps build the reputation of software companies. Though software testers perform a role that requires specific tasks and skills, in-depth studies of software testers lag behind research studies of other roles within software development teams. In this paper, we aim to create a profile of testers by presenting an empirical analysis of the skills the industry currently needs. We analysed data from 400 job ads in 33 countries. We mapped the skills on a taxonomy comprising test-related, technical, and domain-specific skills. In addition, we looked at the demand for educational attainment, relevant certifications, and previous experience requirements.

Our findings show that employers are mostly interested in skills related to test planning and design, test automation, functional testing, performance testing, and progress reporting. One third of the job advertisers were interested in people with the skills to operate test execution tools. Selenium was the testing tool most in demand. The testers must have strong technical abilities, including programming skills in Java, C#, and SQL. Also, they must handle project management tasks such as estimation, risk management, and quality assurance.

Employers do not emphasise domain-specific knowledge, which indicates that they consider testing skills portable across industries. One in seven job ads asks for a software testing certification. Our study helps clarify the complexity of the testing job and outlines the capabilities one needs to fulfil a software tester's responsibilities.

Keywords: software-testing skills, software-testing skill requirements, software industry needs, empirical software engineering, human aspects of software engineering

1 Introduction

The skills of the members of software development teams are crucial to the success of software projects because they directly affect central aspects of software attributes such as performance, reliability, and simplicity [1]. Numerous studies have evaluated the impact of team members' skills on the quality of software produced, the performance of the team, and the competitive market advantages those skills bring (see [2], [3], [4]). Alternatively, as many researchers have shown, having a lack of skilled employees on software development teams directly affects the software projects' costs, their delivery times, and even their completion [5]. The consequences of badly done testing can creep into software development, which often leads to low-quality software. As the tragic series of events involving Boeing 737 Max airplanes demonstrated recently, insufficient software testing can damage products and companies [6], leading to the loss of billions of dollars, logistical nightmares [7] rippling through dozens of large companies, the disruption of business [8], and the loss of people's jobs or even lives.

Analysing the skills of employees, particularly those trained in a specific work environment, is important because the application of those skills enables managers to control and adjust the work process [9]. Skilled team members can also help create and maintain an enriched professional environment that fosters responsibility, the identification of problems and mitigating actions, the spectrum of task assignation, and participative decision-making [10]. Studying the skills expectations for testers is relevant for understanding the nature of the tasks they need to carry out, the importance and the extent of the technical tasks they have to accomplish, and the amount of specialised domain knowledge they require to fulfil their jobs.

Over the years, software developers and their skills have gained attention from both an industry perspective [2], [3], [4] and an educational perspective [11], [12]. However, the same cannot be said about software testers: No earlier than ten years ago, testing was seen as a side-activity, with testers sometimes regarded as second-class citizens whose role was considered a junior- or entry-level position [13]. A recent study by Capretz et al. [14] showed an even more startling picture: Among the people working in software development, tester and maintainer are the least popular roles. They found that the vast majority of engineers do not like testing and would not choose the role of tester [14]. Another study reported that testers in agile projects received less attention than developers in daily meetings, mainly because the Scrum Master viewed the information from the developers as more important [15]. In consequence, for us to look into the job, tasks, responsibilities, and demand for software testers becomes of clear importance.

Currently, "software tester" is a frequently used term but lacks a concrete definition. While "software developer" is well-defined [16, 17], tester is recognised as a specific role without further details. The software testing standard ISO-29119:1 [18] defines tester as "one who develops and tests deliverables and completes the processes associated with the dynamic test process". In the Standard Glossary of Terms, the International Software Testing Qualifications Board (ISTQB) defines testers as "skilled professionals involved in the testing of a component or a system" [19]. None of the most used lexicons [20-23] or the World Standards Cooperation [16, 18, 24, 25] provides definitions or side information on software testers, software testing, testing requirements, testing tasks, testing skills, or testing responsibilities. However, by searching for the same term not related to software, we derive the following: A tester is a person or machine who assesses the quality or state of a thing by testing it [19, 23].

Mathur et al. [26] described testers as the ones responsible for carrying out testing and building up test cases and test plans. Davidov et al. [27] described testers as professionals who identify new defects from failed test cases, analyse defects, and report them in a bug-tracking system. Some literature portrays testers as the development team's worst enemies because they check everything and bring news of defects [28]. However, in teams with high psychological safety, testers are not afraid that developers are offended by the bugs discovered or mistakes they point out that developers have done, but rather that developers are actually happy when the testers report bugs [29]. Even though the descriptions we found are useful, practice shows that testers do much more than the literature describes.

Currently, testers are much better integrated into development teams than they used to be. Software companies recognise them as an important part of the team, especially in Agile environments [30], and the consolidation of Agile teams merits careful attention regarding learning, work, and performance [31].

Another reason to study testers is that we understand intuitively that they do more than testing, and we aim to unveil what else is under testers' responsibilities and how important those responsibilities are for employers. As Dromey demonstrated 15 years ago [32], a preventative approach to software quality is more efficient than a curative one. But this approach entails a number of complex tasks, including design, process management, and adherence to standards. This complexity requires skills to manage it and, in this context, analysing the skills testers need to succeed in their job becomes a necessity.

To increase the understanding of the software tester's role, we extracted its comprehensive profile from the needs the industry currently expresses. In our analysis, we focus on the software tester role instead of on the overall process of software testing. We chose this approach because there is a significant difference between the skill requirements at the whole team level and the skill requirements for the software testers, and we aim to determine traits, discover gaps, follow trends, and make recommendations for the software testers.

Regarding skills, understanding the state of practice and checking whether the testing practice meets the industry's needs is a difficult process [33] that requires scrutinising many aspects. The current state of the industry requires a thorough and consistent evaluated of one's skills at hiring, including inspection of the education and experience listed on the CV and follow-up with interviews and tests [34]. We chose this approach because in the hiring phase, employers need to formulate clear and direct requests for employees' skills that will allow the company to succeed in delivering completed software projects.

We researched two subsets of skills in two previous studies [35], [36]: soft skills demands for testers and the overall testing and tool requirements for testers. The previous studies show that testers need to have good communication, analysis, and problem-solving skills. At the same time, the need for adaptability, openness, and independent-working and team-playing skills has increased significantly. New demands have emerged in the areas of work ethics, customer focus, and the ability to work under pressure. Our current study extends to a large degree the context of the analysis beyond these two skill categories by focusing on an in-depth look at test-related and technical skills and requirements for educational attainment and previous experience, for which we created taxonomies to categorise and analyse the demand.

In our current study, we make a thorough analysis of three kinds of skill requirements for testers: First, we study the testing-related skills demand; then, we study the technical skills that testers need to have; and last, we analyse the domain-specific knowledge that testers need to possess for their skills to be portable when they switch jobs. We also examine testers' required level of experience and educational attainment and other training and certifications necessary to prove their skills.

In this light, we aim to answer three research questions:

RQ1: What skills do software testers need to have according to industrial demand?

RQ2: Are there any correlated skills shaping the profile of the software tester?

RQ3: What educational attainment, certified qualifications, and previous experience do hiring companies ask of software testers?

The remainder of the paper is organised as follows: Section 2 describes the approach we undertook to obtain the skills requirements for a software tester. Section 3 reports the results of our analysis. Section 4 discusses the findings and provides our interpretation of the results. Section 5 presents the implications of the analysis for different categories of audience. Section 6 approaches the limitations of our study. Section 7 draws the conclusions of our findings and suggests future development of the current research.

2 Research method

The current study is an empirical analysis of industrial needs for various skills for software testers from the employers' perspective. To analyse the software tester role, we created a taxonomy for the software testers' requested skills. We chose to create a taxonomy to map the skills required for software testers, as it is a common way to organise knowledge systematically in a domain of interest in which there is little knowledge on a topic [37] and one needs to identify categories, topics, and constructs.

Our analysis is based on a collection of job ads that we processed to allow us to perform both quantitative and qualitative data analyses. The number of job ads we studied gives us a 95% confidence level and a 5% confidence interval for the quantitative analysis. The approach is relevant in that it complements the theory of skill with a hands-on analysis of concrete skill requirements. Additionally, the quantitative and qualitative analysis combination enables us, on one hand, to discover which kinds of skills are in most demand, the level of the demand, and the average number of relevant skills employers are seeking and, on the other hand, to substantiate our findings, mining information to reveal specific traits of the skills as demanded by employers. As we found no significant differences in the number or kind of demands between the ads for the different names of the role, we analysed the entire set of job ads as one.

The structure we used for the testing-related skill analysis is based on the structure of the ISTQB Foundation Level Syllabus [38], and we created a structure for analysing the technical skills. We understand technical skills to refer to any programming, software engineering, project management, or other skills needed in the area of software development that are not typically the responsibility of software testers: the skills that the programmers, software designers, software architects, team leaders, auditors, etc., are more likely to need.

2.1 Data collection

To get an overview of the skill requirements for testers, we collected and analysed 400 job ads published online for 33 countries between December 2016 and February 2017. We chose to use online job-search engines to collect the raw job advertisements instead of going to specific hiring companies. This approach is an efficient way of including in our analysis a large number of employers, a good diversity of job profiles, and large visibility to job seekers.

We investigated the most significant job-search engines through two dimensions: the number of users and the number of jobs posted. According to commercial Web traffic data and analytics services provided by Alexa (Amazon)[1] and SimilarWeb[2], we chose the five most popular job-search engines [39-42].

We collected the job ads from the following countries: USA (96), Canada (65), Norway (22), UK (20), Argentina (17), France (17), Mexico (15), South Africa (14), China (14), Vietnam (13), Greece (13), India (12), Sweden (10), Portugal (10), Australia (10), and others (52).

One of the amenities job search engines offer that we used is that they interpret the search criteria and give relevant results for a range of jobs related to the search term. Therefore, we input the search string "software tester", and we obtained as results job

[1] https://www.alexa.com/

[2] https://www.similarweb.com/

ads for software testers, technical testers, functional testers, QAs, usability testers, performance testers, mobile testers, Web testers, security testers, automated software testers, financial-systems testers, game testers, and so on. Using this tool helps avoid data-collection bias related to the name of the jobs advertised for software testers.

2.2 Data analysis

We collected all the information we found on the job descriptions, responsibilities, requirements for all the analysed ads. We chose this approach, as not all advertisers use the same format for the ads: We found skill requirements for the advertised position in almost all the fields we studied. Next, we cleaned, categorised, and coded the collected information to group it and then analysed it quantitatively and qualitatively. We created the structure for the demands as shown in Fig. 1 and used it for grouping the information we collected.

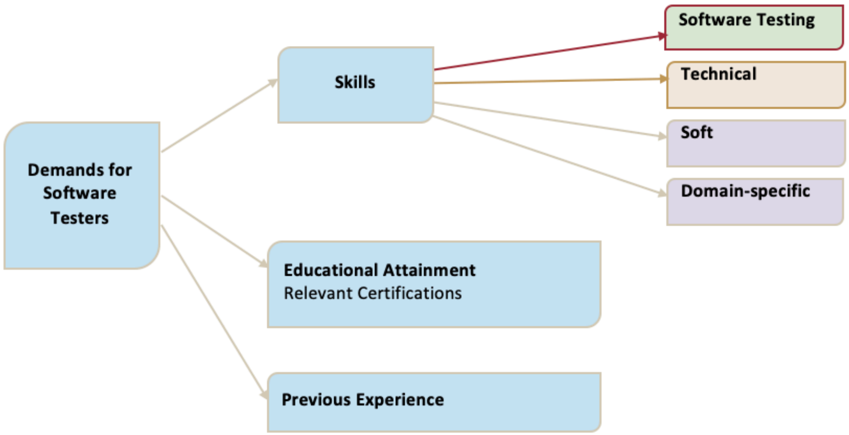


Fig. 1 Overview of the software industry's demands for software testers

To examine the software testing-related skills, we created a separate taxonomy using the structure of the information in the ISTQB Foundation Level Syllabus 2011 [38] as a basis. We chose this syllabus as a reference for our research since it is a representative tool for structuring software testing knowledge. The ISTQB uses is as the theoretical foundation of examining and providing a standardised qualification for software testers. Currently, the organisation has issued over 640,000 certifications in 120 countries [43]. By following its table of contents and selecting from it the parts related to skills, we obtained a subset of the original table of contents that forms the taxonomy of test-related skills, as shown in Fig. 2.

We calculated the percentage of ads and the mean number of skills demanded for each category and subcategory in our testing-skills taxonomy, and where it was useful, we performed a free-text analysis of the frequency of terms to obtain information of interest regarding specific demands. For the free-text analysis, we used an online tool[3] that analyses the frequency of words in large strings and offers options such

[3] https://wordcounter.com/

as inspecting the roots of words, grouping variations together, and excluding insignificant strings, which we used on the collection of coded information that we obtained as a result of the job ad processing. By looking at the categories of software testing-skills demands, we got an overall picture of the skills the industry currently seeks. By looking at the ranked list of subcategories, we determined the popularity and the spread of the demands.

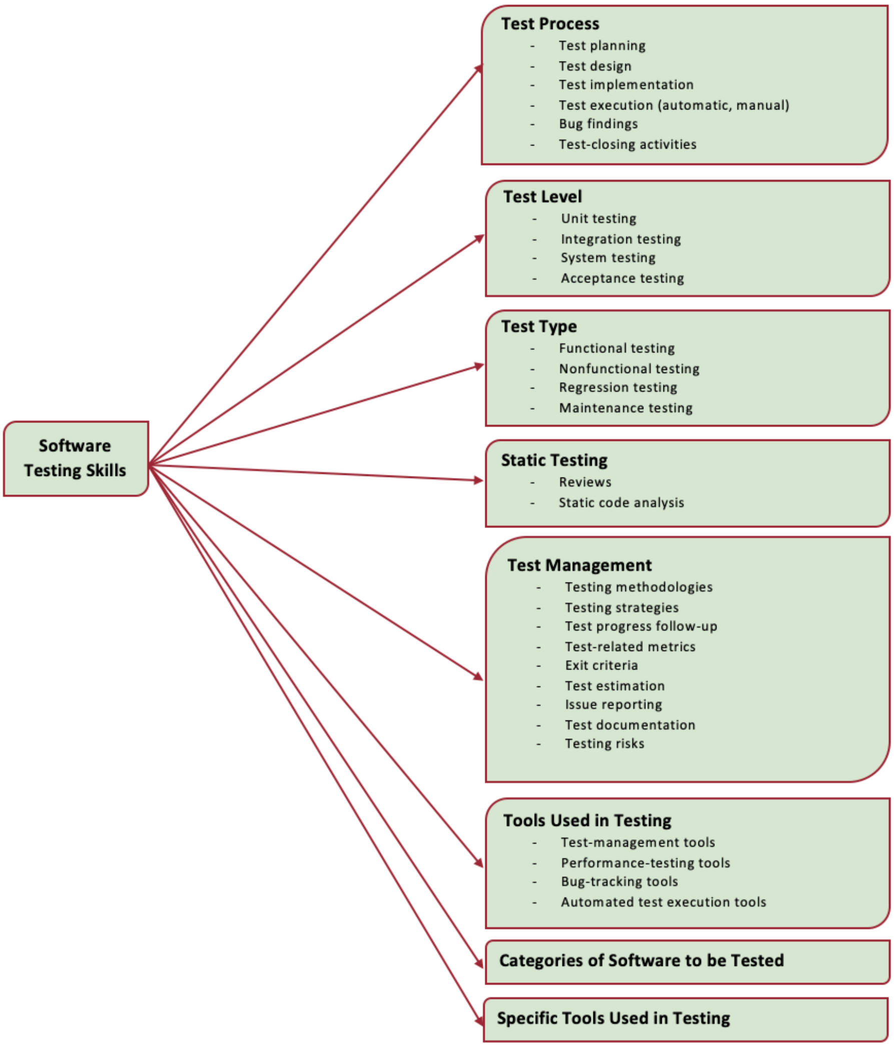


Fig. 2 Taxonomy of the software-testing skills for testers

Through a qualitative analysis, we created a separate taxonomy for mapping the technical skills by labelling and bracketing all the demands related to the technical aspects of development (see Fig. 3, depicting the areas of demand for testers' technical skills).

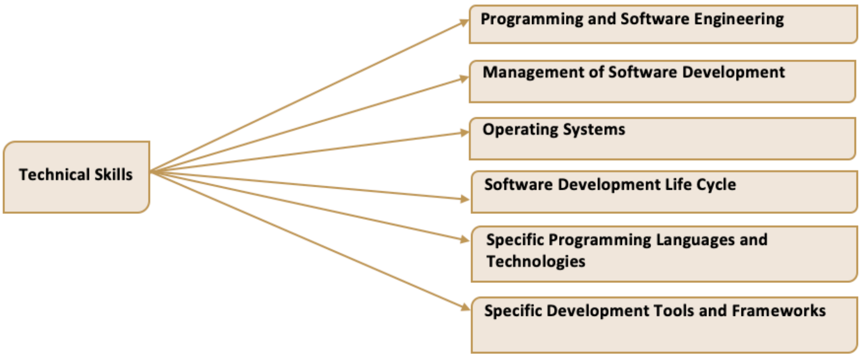

Technical Skills
Programming and Software Engineering
Management of Software Development
Operating Systems
Software Development Life Cycle
Specific Programming Languages and Technologies
Specific Development Tools and Frameworks

Fig. 3 Taxonomy of the technical skills for testers

We did not create a taxonomy related to domain-specific skills. Instead, created a ranking of these skills, in which we used domain specificity to perform a free-text analysis of the information we collected, and we extracted the most frequent terms. We manually went through the results and sorted them again by grouping demands logically belonging together. We proceeded in the same manner for the educational attainment, relevant certification, and seniority requirements. We also performed a quantitative analysis for the minimum number of years of prior experience that employers requested.

For the quantitative assessment, we used SPSS to perform a statistical analysis of the numerical parameters of our data. We obtained the percentage of ads asking for each category of skills, the mean number of skills per category, the maximum number of skills per category, and the number of skills demanded in the lower and the upper quartiles of the ads. The data file available online for access and download [44] provides the details regarding the techniques we used to analyse each category and subcategory of skills.

Moreover, we aimed to discover any patterns in the subcategories of skills being asked for together in the same ad. A strong correlation of skills would have pointed to the existence of various tester profiles that employers are looking for. Inversely, a lack of patterns would have indicated that there no clear software tester profiles emerged from the employers' demands at hiring. To study this aspect of our research, we took into account all of the subcategories of skills in the advertisements and checked how often they appear together in the same ad, two by two. Therefore, we used SPSS to generate a correlation matrix for the identified skills. We looked at the Spearman correlation coefficient, which indicates if there are correlations between pairs of variables. The coefficient's value indicates the nature (direct or inverse) and strength of the correlation.

We searched for significant differences in the requirements for various categories of testers, such as testers and test analysts, and we searched for particular requirements for test managers.

3 Results

3.1 The profiles of software testing jobs

A first finding in the landscape of software testing regards the diversity of names used for the software testing role. Out of all the ads, the top ranked requests were for software testers (77 ads), QA testers (41 ads), automation testers (9 ads), test engineers (7 ads), test analysts (7 ads), manual testers (4 ads), penetration testers (4 ads), and business analysts (4 ads). Out of the 400 job ads, only three asked for test leaders and three asked for test managers. From the remaining ads, more than half (213) used unique names for the software tester role. Examples include "IT tester", "cybersecurity testing", "MS dynamics CRM tester", "senior ETL tester", "testing specialist with HIX/HBX experience", "senior testing automation for network management", "digital validation engineer", "linguistic tester", "QC staff", "algorithms for driver assistance tester", "content producer tester", "technical junior", etc.

Focusing on the skills demand, Table 1 displays the industry's level of demand for all software testers, divided into the four skill categories (Fig. 1). With more than 97% of advertisers and an average of more than 10 skills per ad, the overwhelming majority of employers asked for testing-related competence from testers, followed closely by technical skills. Two thirds of the ads asked for soft skills, and only one third of the ads sought domain-specific skills.

Table 1. Skills demand for testers

Type of skill	Percentage of ads	Avg. skill/ad
Software Testing skills	97.7%	10.2
Technical skills	82.5%	5.4
Soft skills	64.2%	4.8
Domain-specific skills	34.7%	1.7

3.2 Software-testing skills demand for testers

Table 2 and Fig. 4 describe the industry's need for software-testing skills, from complementary viewpoints. Table 2 features a ranking of the categories of software-testing skills and the percentage of the ads asking for them, while Fig. 4 shows the interquartile ranges for the number of software-testing skills in each category.

Most of the hiring ads asked for skills related to the testing process (see Table 2) and, moreover, at least four skills of this kind (see Fig. 4). Three out of four ads asked for test management skills, and testers had to have mastered at least three aspects of it. The ability to perform at least two different types of testing, such as functional, performance, was asked for by more than half of the advertisers. The test tools were another sought-after category of skills, with more than half of the ads asking for testing-tool skills. Most of the ads sought software testers who had mastered about two tools and a little below half of the ads asked for specific test tools. Only one fourth of the ads asking for testing tools asked for competency in five or more tools.

We observed that the request for specific levels of testing was on the lower side, with less than one in three advertisements asking for it and with less than two skills to master.

Table 2. Categories of testing skills demand for testers

Category code	Skill category	Ad percentage
T1	Test process	87.2%
T2	Test management	73.2%
T3	Test types	65.7%
T4	Test tools*	59.5%
T5	Test levels	28.2%
T6	Software categories	25.5%
T7	Reviews and static analysis	12.2%

* 42.0% of the ads asked for specific test tools.

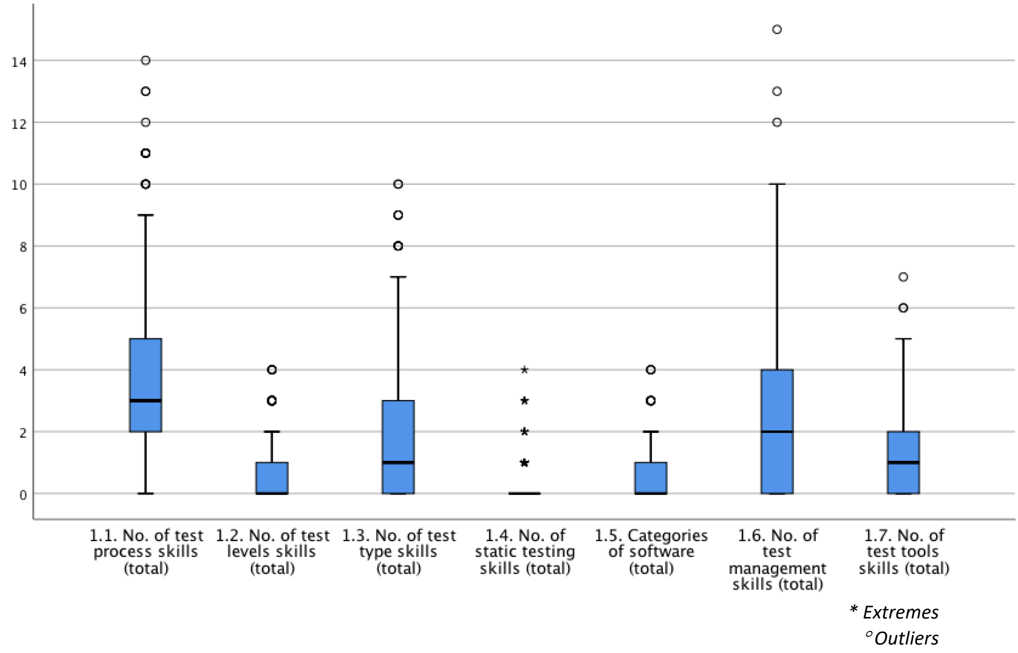


Fig. 4 Interquartile range and means of the software testing skills demand for testers

Going into the details of the skill request, our findings show that most of the advertisers asked for specific skills belonging to the category of test process (T1): planning, design, and execution of tests, whether assisted or not by tools (see Table 3). Next in the ranking is the request that the testers track their own testing through progress reports, bug reports, and follow-up of the bug fixes. The industry is interested in testers with skills in test strategies and methodologies to a smaller extent. Similarly, marginal requests were for the software testers to manage their test environments, write test-related documentation, manage testing risks, handle test data, and handle exit criteria and their corresponding test-related metrics.

Table 3. The hierarchy of subcategories of testing skills demand for testers

Subcategory of skill requested	Percentage of ads
Test design *(T1)*	52.7%
Automated test execution *(T1)*	43.5%
Manual test execution *(T1)*	42.0%
Test planning *(T1)*	40.0%
Automated test execution tools *(T4)*	30.7%
Test progress reporting *(T2)*	29.5%
Follow-up of bug fixes *(T2)*	28.2%
Bug reporting *(T2)*	27.7%
Functional testing *(T3)*	27.2%
Performance testing *(T3)*	23.7%
Test management tools *(T4)*	22.5%
Web applications testing *(T6)*	18.5%
Regression testing *(T3)*	16.0%
Test implementation *(T1)*	14.7%
Bug identification *(T1)*	14.5%
Mobile applications testing *(T6)*	13.5%
Testing methodologies *(T2)*	13.2%
Test strategies *(T2)*	13.2%
Acceptance-level testing *(T5)*	12.7%
Test techniques *(T3)*	12.0%
System-level testing *(T5)*	11.5%
Test environments-related activities *(T2)*	10.2%
Integration-level testing *(T5)*	10.0%
Documentation writing *(T2)*	9.7%
Risk management in testing *(T2)*	9.2%
Interface (API) testing *(T3)*	9.0%
Bug-tracking tools *(T4)*	8.2%
Test data management *(T2)*	7.7%
Unit-level testing *(T5)*	7.5%
Test-related metrics *(T2)*	7.0%
Performance-testing tools *(T4)*	7.0%
Black-box/grey-box testing *(T3)*	4.7%
Exit-related criteria *(T2)*	4.5%
White-box testing *(T3)*	4.0%
Maintenance testing *(T3)*	2.0%
Test effort estimation *(T2)*	1.7%
Desktop applications testing *(T6)*	1.7%
Large-scale applications testing *(T6)*	1.5%
Client-server architecture testing *(T6)*	1.2%
Test closure activities *(T1)*	1.0%

The results show that testing the functionality of software products was the most sought-after type of testing skill (T3), followed by the requirement for performance testing skills. A relatively small number of advertisers from the industry asked explicitly for regression testing or maintenance testing.

Regarding software testing on different levels, the industry seems to focus on high-level testing, with acceptance testing and system testing occupying the first two positions in the ranking of test-levels skills. The employers asked one in ten testers to perform integration testing. Unit testing was not in high demand for testers.

Most of the ads did not ask for skills in testing specific structures of software, such as distributed applications. Less than one in five ads asked for skills in testing Web systems, and one in seven ads asked for skills in testing mobile systems. About half of these ads asked for both mobile and Web competency. Besides these requests, other demands were rare, with very few advertisers needing skills for testing large-scale systems, client-server architecture, multi-interface systems, or database systems.

The job advertisers were interested in hiring people with the skills to operate tools in the category of automated test execution. The tools in the category of test management tools were demanded to a slightly smaller extent than the test execution tools, with one in four employers looking specifically for them. One in twelve advertisers asked for competencies in using tools in the category of performance-testing.

Table 4. The hierarchy of specific testing tools demands for testers

Testing tools	Tool category	%of ads
Selenium	Automated test execution	18.0%
Jira	Test management, bug tracking	10.2%
HP Quality Center	Test management, bug tracking	8.7%
QTP	Automated test execution	5.7%
JMeter	Performance	5.2%
LoadRunner	Performance	4.0%
Cucumber	Automated test execution	3.7%
SoapUI	Automated test execution, API testing	3.2%
Confluence	Test management	2.7%
TestNG	Automated test execution, unit testing	2.5%
JUnit	Unit testing	2.2%
HP ALM	Test management	1.7%
Microsoft Test Manager	Automated test execution, Test management	1.5%
Fitnesse	Automated test execution	1.2%

Looking into the details of the demand for tools, the most asked for was Selenium, a tool in the category of automated test execution. The second most requested tool, Jira, is one of the most commonly used tools for test management [45] and comes with comprehensive bug-tracking systems and a myriad of add-ons for project management, time tracking, and integration. HP Quality Center, the third of the ranked tools requested, is a test execution tool that is part of the Application Lifecycle Management solution offered by Hewlett-Packard. QTP, the fourth most requested tool, is also a popular automated test execution tool for functional testing [46]. JMeter and LoadRunner are both performance-measurement tools that used to be the exclusive responsibility of developers [47] but are currently widely used by testers as well [48].

3.3 Technical skills demand for testers

Table 5 and Fig. 5 describe the industry's need for technical skills from complementary perspectives. Table 5 ranks the categories of technical skills demand for testers and the percentage of the ads asking for them, while Fig. 5 shows the interquartile ranges for the number of technical skills in each category.

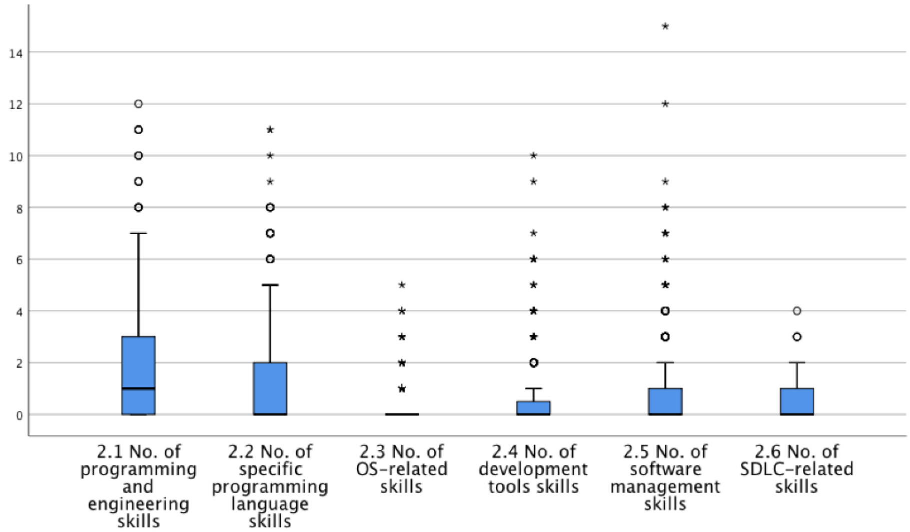
Two thirds of the ads want testers with programming and software engineering skills, while almost half of the ads need the testers to master specific programming or scripting languages (Table 5). However, our results show that testers are asked to be proficient in no more than two programming languages (see Fig. 5).

One third of the ads asks for something related to the management of software development, and another third of the ads asks for a specific development life cycle to be followed.

One in seven employers is interested in testers with specialised knowledge on a particular operating system (OS; Table 5). Requirements to master more than one OS are rare (Fig. 5). One in four employers ask the testers to use specific development environments or tools. When the testers are asked to use these, in the majority of cases, they have to use just one development tool or development environment.

Table 5. Categories of technical skills demand for testers

Skill category	Percentage of ads
Programming & Engineering	69.0%
Programming Languages or Technologies	42.2%
Software Management	36.2%
SDLC-related	33.2%
Development Tools and Systems	25.0%
Operating systems-related	14.5%



** Extremes*
° Outliers

Fig. 5 Interquartile range and means of the technical skills demand for testers

As Table 6 shows, most of the demand is for programming languages or frameworks, but a significant number of ads ask for the following software engineering skills as well: continuous integration, deployment, web services, or networks. The only development technique that is specifically asked for is test-driven development (TDD), but this request is made in rather low proportions (3%).

From the programming languages in demand, SQL takes first place with one in five ads asking for it. The second most demanded language is Java. Third place is taken by C#, which has half of the demand that Java does. We observe, therefore, that the testers are currently being asked mostly for compiled languages. The next two languages in demand are dynamic: JavaScript closely follows the proportion of demand for C#, and Python falls only slightly behind JavaScript. Next in demand is C++. HTML, XML, Ruby, C, CSS, and Perl are close and rather low in demand.

In software management, the competency with the most demand is the quality assurance (QA) of software systems and the processes implementing it. Next in demand is the request for skills in project management and requirements management. While there is some demand, testers are not necessarily asked to be business analysts at the same time, with only over 1% of ads demanding such skills.

Regarding various software development life cycles that software companies adopt, the most demand is by far for Agile. Only a fraction of employers asks for their testers to master Waterfall, Kanban, or V-model.

With focus on the specific development tools or environments, most of the demand is around deployment tools, such as Jenkins or TeamCity. Build tools, such as Maven, are also demanded, as are classic source-control systems, such as GIT or TFS. SharePoint was one of the few online collaboration frameworks specifically asked for and had rather low but notable demand. Only a limited number of jobs require testers to have skills in the use of specific tools or systems with regards to software development. The top request, with one in twenty ads asking, is for Jenkins, a tool used for continuous integration and continuous delivery.

Related to operating systems, most in demand seems to be desktop OSs and Unix/Linux OS. The third in the ranked requests is Windows. Last are mobile OSs, Android being most demanded and iOS following it. No other mobile OS was specifically asked for.

Table 6. Sub-categories of technical skills demand for testers

Technical skills requested	Percentage of ads
Programming & Engineering	
Programming tasks	42.2%
Various technologies	13.0%
Continuous integration	6.7%
Various frameworks	6.2%
TDD	3.0%
Web Services	2.5%
Networks	2.0%
Continuous delivery	1.7%
Programming Languages or Technologies	
SQL	20.2%

Java	14.5%
C#	7.7%
JavaScript	7.2%
Python	6.7%
C++	6.0%
HTML	3.5%
XML	3.2%
Ruby	3.0%
C	2.7%
CSS	2.2%
Perl	2.2%
Software Management	
QA processes	12.5%
Project management	7.5%
Requirements management	7.2%
Business analysis	1.5%
SDLC-related	
Agile	31.2%
Waterfall	3.2%
Kanban	1.2%
V-model	1.0%
Development Tools and Systems	
Jenkins	4.2%
GIT	2.2%
TFS	1.7%
Maven	1.2%
TeamCity	1.2%
SharePoint	1.0%
Operating Systems-related	
Linux	9.5%
Unix	5.0%
Windows	3.7%
Android	3.5%
iOS	2.7%

3.4 Domain-specific skills demand for testers

The demand for domain-specific knowledge covers a wide variety of industries. Out of the one third of the ads that ask for domain-specific competencies, most of the demand is around financial services (4.7%) and banking systems (3.2%) (see Table 7). Most of the other notable requests are in financially connected areas, such as payment, e-commerce, and accounting. Next are requests for the telecom industry (4.5%) and logistics (1.2%). The request for skills in gaming software is at around 2% among all job ads, while the other ads (30.7%) cover a large spectrum of other kinds of software.

Table 7. Domain-specific skills demand for testers

Domain-specific skills requested	Percentage of ads
Financial services systems	4.7%
Telecom	4.5%
Banking systems	3.2%
Gaming software	2.2%
Insurance industry	2.0%

Payment systems	2.0%
E-commerce	1.7%
CRM	1.5%
Automotive	1.5%
Networks	1.5%
ERP	1.2%
Accounting	1.2%
Healthcare	1.0%
Others	30.7%

3.5 Correlated skills

To identify patterns in the profiles of software testers that emerge from the employers' ads, we checked for correlations between the different skills asked for in the advertisements. Our sample size of $n = 400$, where we included all the testing-related, technical, and soft-skill requirements for testers, was grouped by using the taxonomy in Fig. 2, Fig. 3, and study [35]. We split the data in two sets to look for correlations among the high level of skills (the first level of nodes in our taxonomies) and to look for correlations in the details of the skills (the leaf nodes of the taxonomies).

We used the Spearman correlation coefficient (ρ) for each pair of skills, as it shows correlations between variables linked by any monotonic function, not just by a linear relationship. At the same time, the Spearman coefficient is appropriate for ordinal variables, the form in which our skills-related data are stored. The resulting Spearman coefficients can be downloaded on the Web: [49] (high-level skill analysis) and [50] (low-level skill analysis). We used the interpretation of the Spearman correlation coefficient in psychology given by Dancey & Reidy [51].

a) Correlation analysis on the high level of skills

In the analysis of skills at a high level, we found a set of relevant correlations, which are presented in Table 8.

Table 8. The correlation of skills analysis, high level

Skills	**ρ values**
Strong Correlations	
Technical & programming	0.79
Testing & testing process	0.77
Testing & test management	0.74
Moderate Correlations	
Test management & test process	0.57
Testing & specific types of testing	0.49
Technical & software management	0.49
Testing & test tools	0.48
Min. years of experience & other test-management activities	0.44
Technical & operating system knowledge	0.42
Technical & domain specific	0.39
Testing & test levels	0.38

Testing & software management	0.38
No. of responsibilities & min. years of experience	0.33
Testing & soft skills	0.30
Testing tools & specific software development life cycle	0.30
Inverse Correlations	
Min. years of experience & education attainment	-0.23
Min. years of experience & soft skills	-0.16
Test tools & domain specific	-0.11

When the employers have technical demands for testers, they will most likely be related to programming and then related to using various OSs. But there is also a significant connection between the technical skills that the employers look for in testers and software management skills and domain-specific skills. Part of the technical testers are likely needed to take responsibilities from the project management area and to hold domain-specific knowledge. However, it is likely that they will not be involved in requirement analysis and manual testing.

A second result is the strong connection between the testing skills and testing progress and test management skills. The data show us that employers strongly opt for testers with a grip on the entire testing process, and the ability to manage testing. Many employers seem to ask together for bug-tracking skills and testing related to changes.

Part of the domain-specific testers are asked to have the ability to operate testing tools, but they are not likely to be asked to design automated tests, to do white-box testing, or to have performance-testing skills.

The more experienced the testers are, the more likely they will be asked to perform other miscellaneous activities related to test management. In addition, the more experience is required from the testers, the more responsibilities and tasks it is likely that they will get. Our results show that experienced testers are not likely to be asked for education attainment nor for specific certifications. A surprising result shows that it is not likely that experienced testers are asked for soft skills.

b) *Correlation analysis on the low level of skills*

In the analysis of skills at a low level, we found a relevant set of correlations, which is presented in Table 9.

Table 9. The correlation of skills analysis, low level

Skills	ρ values
Strong Correlations	
Automated test design & automated test execution	0.71
Moderate Correlations	
Black-box testing skills & white-box testing	0.61
Specific testing tools & automated testing tool	0.58
Specific testing tools & test management tools	0.51
Manual test design & manual test execution	0.51
Organisational & analytical and problem-solving	0.40
Fast learning & interpersonal	0.38
Organisational & ability to work independently	0.38

Communication & organisational skills	0.38
Non-functional & functional testing	0.37
Ability to work independently & problem-solving	0.35
Fast learning & openness and adaptability	0.35
Openness & organisation	0.34
Automated test design skills & automated test tools	0.33
Bug racking & testing related to changes	0.33
Analytical & team player	0.30
Team player & test metrics	0.30
Inverse Correlations	
Programming language & requirements analysis	-0.14
Domain specific & performance testing	-0.12
Programming tools & manual testing	-0.12
Development tools & manual testing	-0.12
Test techniques & fast learning	-0.11
Domain specific & automated test design	-0.10
Domain specific & white-box testing	-0.10
Openness & adaptability & test strategies	-0.09
Testing tools & manual testing	-0.09

The correlation analysis results show the following: the employers ask for testers who can perform both the design and execution of tests. Part of the employers ask the testers who perform white-box testing to perform black-box testing too. In addition, the correlation analysis confirms the quantitative analysis that when the employers ask for testing tools, they will most likely ask for automated test execution tools, test management tools, and some other specific testing tools. They are less likely to ask for specific bug-tracking tools.

The domain-experts are not likely to be asked for performance testing, design of automated tests or white-box testing. The more technical skills the employers look for in software testers, the more likely they will not ask for abilities with manual testing.

Regarding soft skills, we observe a tendency among the employers to ask testers to have the ability to work independently while having problem-solving and organisation skills. These employers associate fast-learning skills with interpersonal skills and the openness and adaptability skills in their demands. Employers also seem to associate fast-learning skills with mastering various test techniques, such as exploratory testing, use-case testing, or boundary-values testing.

Another notable result is the correlation that the employers make between the team-playing skills and the ability to define and use test metrics.

3.6 Education attainment and certifications

The greatest number of ads (85%) ask for completed bachelor studies. Only one-tenth of the ads ask for more, such as completion of master-level studies. While there is a request, the number of ads asking for PhD studies is negligible (0.5%).

Most employers do not require specific certifications related to testing. From the 14% who ask for such certification, the majority of the requests are for ISTQB Foundations-level Tester (9%) and ISTQB Advanced-level Tester certifications (1%). A negligible number of ads (<1%) ask for Application Life Cycle Management (ALM) certifications, information systems security-related certifications (CISSP), and network penetration testing certifications (GPEN).

3.7 Previous experience

About half of the ads (54%) ask for previous experience. The calculated average of years required from these ads is 3.4. With less than 1%, only a few exceptional ads ask for 10+ years of prior experience. After removing the exceptional cases, the majority of ads ask for two to three years of previous relevant experience. Most of the job ads (83%) do not ask for specific seniority; only 10% of the ads ask for senior testers, and 5% of the ads ask for juniors.

We found that test managers are asked for nine skills on average per advertisement. Looking into the details of the demand, the first request is for test managers to perform software testing. Second comes the demand for test management skills, test planning skills, and skills with automated testing tools. Only two out of the six ads ask the test leaders to manage testers, to preform test estimations, and to monitor the test's progress. The specific requests in the area of tools is varied, and there is no strong request for a particular tool.

4 Discussion

Our results show that software testers need to be highly skilled in their job and that they need to master a wide range of skills. By looking at the spread of the demand, testers are not asked to be specialised in one certain activity, such as performance, automation, or test management; instead, they are asked to master many and various aspects of testing. We infer that most testers are asked to play an active role throughout the testing process. By analysing the landscape of job names used for the role of software testers, we conclude that the employers place the skills essential to them in the title of the job they advertise for, such as the knowledge of a tool (Jira, Selenium, CRM) or a class of testing skills (functional, network, penetration).

Aside from this, most employers ask testers for a wide span of technical skills. In regards to testing, the technical requests are not specialised on narrow aspects; they vary from programming skills to development environments and tools, as well as integration skills, deployment, project management, development models, and many others. This finding is in line with an observation from 2010 [52], which found that companies see the technical knowledge of testers as important and more significant than domain knowledge. The finding is also supported by [32], stating that the best approach in software development would be extensive consultation with domain experts.

In a previous study in which we included developers [36], we found that only half of developers need to master anything related to testing and that they are asked to have an average of 2.5 testing skills; the result denotes that testers need to master development better than developers need to master testing. Therefore, developers remain highly focused on programming and software building, while testers are asked to acquire more technical skills and fulfil more programming, software engineering-related, or software management tasks.

4.1 Testing skills, technical skills, and domain-specific skills

In this section, we will discuss our first research question: "What skills do software testers need to have according to industrial demand?"

a) Designing tests is the main demand

The most important software-testing skills for the industry seem to include the ability to design tests. The result reveals that employers want testers to be able to extract the testable parts of software and design corresponding high-level and low-level test cases from the product documentation or other available sources. The requirement to design tests offers a side benefit: a way to evaluate the testability of software, in which the goal is to have a more testable software. Also, through designing tests, testers give flexibility and structure to the testing initiatives: tests are grouped in families, are given ranks and criticality levels, and are categorised as main-stream, edge-scenario, or failure cases. This makes the testing process capable of changing with regards to deadlines, prioritisation, and stakeholders' feedback. The survey [53] finds that design quality is vital if software is to be maintainable and finds a relationship between design quality and testing practices. Therefore, our results support this survey through the employer's demand foremost for design skills, which also encompasses the need for maintainable software, good design, and good testing practices.

b) Automated testing is more asked for than manual testing

The advertisers put more focus on automated tests than on manual tests, and we assume that the higher level of request for automated execution of tests involves more complexity [54] and therefore requires more skills than the manual execution of tests. Moe et al. [55] report that automated testing, an important enabler for Agile development, heavily depends on the knowledge of the testers. They found that most testers still do manual testing, and they consider that it takes too long to update the automated tests. Our finding confirms the continuing need for skills in writing automated tests, reflecting the persistence of this industry gap.

The high request level for test automation tools can be explained by the popularity of Agile development, where working in iterations and regression testing is essential; hence, a significant part of regression tests is automatized. Automating regression tests is the leading approach for maintaining and consolidating the quality of software systems. It is an ongoing process that requires knowledge on automated testing and corresponding tools [56]; consequently, we expect to see a constant rise in this type of demand in the future.

We also expect a clearer allocation for various test automation tasks, such as a behavioural consequence of not having it be shift blame between developers and testers, as reported by Wiklund et al. in their systematic literature review [57]. We find it surprising that only one in five job advertisers asks for test management tools. This misbalance may indicate a tacit requirement for test management tools skills.

The high request for Selenium does not come as a surprise, given that it does not require very technical skills to operate it, it is open-source, and it is portable through all major OSs and web browsers [58]. The high request for other specific test tools confirms the employers' need for testers to perform tool-assisted work and that there is already a tool structure in place and waiting to be used, while the performance-testing tool demand also confirms the technical component sought in the profile of software testers.

c) A common demand is test planning and implementation

The requirement for skills in test planning signals that a significant number of testers are needed to actively contribute to high-level testing. Test planning means determining the scope and the objectives of testing; establishing the test approach; determining the test resources needs (human and logistic); determining the test schedule; establishing test metrics; identifying the potential risk; and mitigating actions, start/stop, and exit criteria. We infer that the testers are needed to manage this aspect of testing as well, as should do so above the typical testing tasks of creating and executing tests.

Test implementation assumes the ability to group the test cases into execution suites and sequence their execution for both manual and automated tests. Babinet and Ramanathan identify three major challenges in large-scale development: the unpredictability in feature development, conflicting priorities, and overlapping release cycles [59]. In this light, we consider that our findings confirm and show the industrial need for a clear prioritisation of tests.

d) It is not common to ask for bug identification, test documentation, test data, or test closure activities

We obtained two surprising results for the two least-demanded skills: bug identification and test closure. Even though one of the most important aspects of a tester's job is to find bugs, few advertisers explicitly ask for it. An explanation is that it is a tacit requirement and is therefore not explicitly set in job advertisements. The other surprising result is that only 1% of the ads ask testers for test-closure activities. This activity assumes checking what deliverables have been delivered, archiving the test ware and documentation, checking that all the project issues and bugs are either closed or deferred, ensuring test-related information is handed over to other departments, and analysing lessons learned for the future. A possible explanation for the lack of demand is that hiring companies do not focus on the activity itself. This result is also confirmed by Kassab et al. in a report on the state of practice in software testing, where closing the test process is not listed as a method or practice in testing [60]. It is also likely that employers assume that it is only the responsibility of project managers, as assumed in a study from 2014 [61].

We expected the industry to give higher importance to test documentation, test data, and test metrics. These three aspects of testing need to be mastered by all the testers within a team. It is not sufficient that only one tester in ten writes test documentation and one in twelve creates and maintains test data. We state this because team members see documentation as a useful and important communication tool [62], even if it is outdated [63], while the lack of test documentation can generate issues around the understanding of undergoing testing or the reproducibility of tests.

Improper test data can be costly because it can either break laws and regulations, such as GDPR [64], while oversimplified test data can hide serious product issues, especially in the performance area. The test metrics are closely related to test management and the predictability of testing; therefore, the lack of requests for it can flag a potential shortcoming in the employer's skills demand.

Even though there is a low demand for risk management and exit criteria, these may be aspects that fall mostly under either the test manager or the project manager's responsibility. Hence, there is no explicit need for testers to master this area.

e) The software testers need to follow-up on bugs to reach a resolution, but the tools to use in bug tracking are unimportant

The high request for following-up on bug reports is likely to be generated by consequences to the testing budget in case the follow-up is not done. The explicit request for bug reports can be an efficient way to formalise the testing process so that all the work around bugs is visible and budgeted. A reason for the request could also be that most software development implies managing a large number of bugs on a daily basis, and fixing them is one of the most common and time-consuming activities for developers [65]; therefore, this activity needs to be measured and planned in detail.

Another surprising result is the low request for bug-tracking tools. The fact that only one in twelve job ads ask for it does not mean that one in twelve testers should use bug-tracking tools: it most likely means that testers use these tools and workflows with standard users' rights, much like they use their email client. The result can be explained by the fact that a significant number of companies have bug-tracking tools that are set-up and maintained by a separate team-handling infrastructure. It is also likely that team members use the same bug-tracking tools as end users who report bugs, which indicates that learning how to operate these tools should be straightforward so that no or very few special skills are required.

f) There is a high interest in acceptance testing and a lower interest in integration testing

We expected that employers would ask the testers to have more skills in API testing, especially since our previous study [36] shows that 9% of the developers are asked to have this skill. We expected that transitioning software towards micro-services and cloud-architectures and the emergence of software supporting the Internet of Things (IoT) would have a greater footprint regarding the demand for all three kinds of testing, especially since the IoT exhibits a large number of resources, connectivity, and interoperability issues [66].

The most significant industrial demand seems to be placed on high-level testing, with acceptance testing and system testing occupying the first two positions in the ranking of skills for test levels. The result concerning acceptance testing is in line with our previous finding on soft skills [35], where there is a tendency towards asking testers for more customer focus. It is also in line with the fact that the lack of user acceptance has long been a major impediment for the success of new software systems [67]; hence, employers currently put more focus on it. Both these levels of testing assume that testers have a good grasp on the overall purpose and functionality of the software being tested.

We expected, however, to see more requirements for distributed software because testing web systems assumes testing different characteristics and doing so with different tools than mobile systems or desktop systems [68]. Therefore, the lack of specificity in this area is most likely a lack in the industry's demand.

Given that review skills rank the last in the testing-related demand formulated by the industry, we consider that testers are still not involved enough in the review process. A significant part of the cost issues and defect sources are found in the design phase. Our result confirms Lee's finding that, of all testing activities, designing tests is practiced by the fewest companies [69]. The fact that only one in eight testers is asked to perform reviews is concerning.

It is natural for testers not to be asked to have static analysis skills. Even though it is a form of testing, static analysis is performed with dedicated tools directly on code by developers and architects; it is done mainly to treat symptoms in the code that would facilitate defects and failures.

g) The software testers have to undertake software development and performance testing

An interesting result is the demand for testers to have performance test tools. Performance checking, which used to pertain exclusively to developers, is now asked of testers, too. If we compare this with the same requirement for developers, it appears that the demand for these skills is slightly higher for testers than for developers [36]. The finding indicates a shift in responsibilities for performance testing towards testers, divergent from the common practice two decades ago [47].

Moving our focus towards technical skills demand, the advertisers' ned software testers to write their own SQL queries when performing testing, therefore, the highest request for this skill. The result is can be explained by two essential aspects of testing: the test data and the verification of the results for the execution of tests. It is a common practice to store data in databases [70]; therefore, the test oracles and the test execution results are to be found in such structures. Nonetheless, databases are heavily used in automated testing, reflecting the need for testers to have query-language-related skills.

Not surprisingly, the demand for testers to manage parts of project development comes as an extension for the test-management request. We observe that, apart from the industrial request for a technical profile, testers also have to possess strong managerial skills. Project management is a step up from test management, but it has many similar traits with [71], such as prioritisation, estimation, resource allocation, risk management, defining and following-up metrics, starting, stopping, and resumption criteria. Therefore, it is natural to ask testers to take responsibility for aspects of managing the project, given that their skills in test management are transposable to project management.

h) The software testers need to be quality assurers and project managers and act as customer support

In addition to the classic testing-related requirements corresponding to their role, a significant proportion of employers ask testers to master knowledge in either the QA field; business requirements elicitation and handling; customer support; or project management, such as risk, release, and resource management. Most of the demand is towards QA, and it is not coincidental that even though the nature of the two job profiles is different [72], many companies merge the roles of testers and quality assurers into one profile. While testers need to check the conformity of the developed software with the software's requirements, determine fit for use, and aim to discover bugs, the main responsibility for QAs is to establish, control, and maintain quality criteria for the software under development and to make sure that the internal and external quality standards are being followed and implemented.

Also, it is natural to involve software testers in the incipient phase of requirements management. As Nuseibeh and Easterbrook report [73], better modelling of problem domains, as opposed to the behaviour of software, is a key factor for the development of successful software systems. Testers play an important role in understanding the business side of the software that is to be built and following it up through the software-development process. Requirements management is also an extension of the tester's job to analyse the testable parts of system requirements, to group them, and to prioritise them. Therefore, asking testers to handle the elicitation and design of business requirements, transform them into system requirements, and follow them up with relevant stakeholders is another case of early defect prevention in which the testers' skills are reused at the project level.

However, testers are not asked for too much on the business side of the companies. They are not asked to assess or handle the business model of the hiring companies nor to recommend software solutions to meet the business's needs. The lack of requirements in this area is understandable in that no test-related skills can be reused in the area of business processes.

Regarding development models for software, a recent study [74] states that 70% of all development projects follow the Agile development model. Combining it with our results, it seems that Agile has become such a widespread practice that the need to specifically ask for it is no longer relevant. Therefore, employers do not frequently ask for Agile-related skills; instead, they just assume testers have them. At the same time, the low request for sequential development models signals that Agile has become the principal state of practice across industries.

i) There is a strong preference for technical testers

With two out of three ads asking for technical skills, we can state that there is a clear preference for testers with technical abilities. Programming and scripting languages are a very common request, along with the request that the testers be familiar with the various approaches in the development and deployment of software. These kinds of requests add a significant dimension of complexity to the profile of software tester.

Moving our attention to software development tools, we note that Jenkins and similar tools for continuous integration are commonly asked for. The most likely reason for this is that these tools are not tailored solely for developers but are part of operating a continuous integration model [75], common in Agile and used across multiple roles. We observe, therefore, an industrial preference for testers with competencies in the technical areas of integration, deployment, source-control management, and build management. These requests for skills are currently a rather common demand within development teams [76]. On top of this, a common practice is that testers use the same tools as developers to trigger new builds or to deploy builds in the testing environment. Ideally, all these activities need to be done without developer assistance, resulting in the demand for such skills. Still, the level of the demand is not high, and an explanation for it is that these tools are easy to operate for standard usage; therefore, specific skills are not necessary.

It is natural to ask more competencies for Unix/Linux-based OSs than for other OSs, as they are highly configurable, exposing several thousand configurable features on over two dozen architectures [77]. It is unclear why the low demand for mobile OSs exists. It would have been reasonable to see more demand in this area, as testing on mobile devices with different user profiles, performance measurements, access rights, and security settings are all common. A possible explanation is that testing is done on preinstalled OSs with standard settings, with the possibly of only minor changes being made to it. It is unclear whether the lack of testing of software with various configurations of OSs is not being done because of a lack of time, lack of resources, or lack of specifications or needs in customer contracts or user agreements.

j) Domain-specific knowledge is a less important request

With one in three ads needing testers who have knowledge in domains other than software, this request is not negligible. However, reading the result in reverse, we notice that two thirds of employers do not ask for domain-specific skills. We can thus state that the employers search for testers with testing skills more than personnel who are specialised in their industrial profile. Most ads asking for domain-specific knowledge pertain to the area of financial services software (online transactions, banking systems, payments). The request can be explained by the high number of rules and regulations that financial services [78], including software, have to be compliant with, therefore assuming testers need the skills related to testing it. Seldom do the telecommunication, software, gaming, and other industries ask for domain-specific skills. There were no significant requests from other branches of these industries, such as infrastructure, governance, IT, health care, tourism, agriculture, and defence. From this point of view, we attain the result that software testing is a role considered to be portable across industries by employers.

4.2 There is no set profile of the software tester, but there are correlated skills in the employers' demands

In this section, we discuss our second research question: "Are there any correlated skills shaping the profile of the software tester?" The two correlation matrixes show that there are no set and clear patterns in the software tester profile emerging from employers' hiring ads. However, we discovered four interesting correlations.

Looking into the details of the request for skills, the result implies that employers prefer software testers who master the testing process in its entirety from conception to completion, not fragments of it. The connection between bug-tracking skills and testing related to changes is most likely due to the following: the more software changes, the more likely is to find a more bugs and, therefore, to have a formal way to follow-up on them from find to fix.

The correlation between technical competencies and software management tasks is important, as it signals that even though employers prefer technical testers, they do not ask for *purely* technical testers, for which it would be sufficient to have the ability to program and manage tools.

Regarding the domain expertise, our findings point to a profile of a domain specialist who is in touch with the technical aspects of development, but will most likely not work with programming, debugging, or carry load and stress tests.

Regarding the accomplished testers, experience plays an important role during hiring, especially as a replacement of formal education and other certifications. However, it remains unclear why employers tend to ask for fewer soft skills from more experienced people.

Focusing on the soft skills, we observe that the employers who need their software to be tested in multiple ways ask for fast learners. This suggests that the various techniques of testing, such as code analysis, use-case testing, boundary values, dry runs, and so on, are time consuming and require significant amounts of effort to master.

An interesting result is the link that the employers make between team-playing skills and the ability to define and use test metrics. More precisely, by looking into the ads, we found that employers need the testers to define metrics, to collect them, and to follow-up on the metrics and the test coverage. Our finding confirms the point in paper [79], which depicts communication as a mandatory part of the metrics life cycle. Employers need to educate the testing team about which information has to be captured to process the metric. Therefore, this is a likely explanation for the correlation that emerged from the employers' hiring advertisements.

The values of the Spearman correlation index signify that there are no clear predetermined career paths that software testers can take. Moreover, there is no predominant or strongly correlated skills demand that depends on the job title being advertised (tester, test analyst, test manager), nor on the seniority of the position (junior, middle, senior). We can infer that the job description and, consequently, the skills demand is determined by the everyday needs of a company or the immediate needs of a project rather than a standard distribution of responsibilities in software testing. As paper [33] shows, there is no one practice in software testing, and the same seems to be true when hiring software testers.

4.3 Seniority, previous experience, educational attainment, and certifications

Regarding our third research question, "What education attainment, certified qualifications, and previous experience do hiring companies ask from software testers?", formal education does not seem to be a focus of employers when they hire software testers. We notice only a moderate demand for testers to have graduated from a certain education program; only half of the ads explicitly ask for a degree from this program. A negligible number of ads ask for the completion of high school studies. The vast majority of the ads asking for education attainment request a bachelor's degree in computer science or a related domain. Very few of these ads ask for a completed master's, and even fewer ask for PhD studies. By comparing the level of request, we can affirm that skills are in more demand than educational attainment.

Certifications are a common way to complement formal education. However, the industrial demand for such documents is still on the low side compared to classic university diplomas; one in seven hiring ads asks for certain certifications within software testing. By one order of magnitude, the most popular demand is for the ISTQB Foundation Level Tester certification, followed by the ISTQB Advanced Level Tester certification [80]. Very few ads ask for certified knowledge in security, penetration testing, or application life cycle management. It seems that the employers usually want the testers to have general knowledge of the testing field. A small number of specialised jobs ask for other software testing certificates. The certifications are demanded in addition to the requirements for completed studies. However, we expect an increase in the demand for training and testing-related certifications. The study [81] performed by Garousi and Junji in 2016 points in the same direction by showing an increasing attention given in Canada to certifications and training on software testing.

Seniority-related requests do not seem to be a focus of employers. This finding supports a study from 2010 [82], which finds that the software development experience cannot replace the demand for a systemic identification methodology and that experienced testers were not necessarily better than inexperienced testers in every aspect.

5 Implications

This paper contributes to the broader field of skills study with the profile of the software tester as currently asked for by the industry. For this specific profile, we distinctly show the type and level of demand for test-related, technical, and domain-related skills. In addition, we show the principal requirements that the employers have for test managers in terms of skills. We also provide insight regarding the type of education required, the certifications needed, and the level of experience sought after by the employers. To provide this information, we have developed and used two taxonomies to group the testing skills and to group the technical skills demanded of testers.

In line with the study by Capretz et al. [14], which notes that software testers need to choose from an immense range of possibilities and that they need to maintain a high level of attention to detail, our results show that the testers need both broad views and attention to detail. Therefore, it is likely that this multitude of requests is vast in nature and requires the expertise of many different kinds, which can be a reason that there are few people, indifferent of the personality type, who prefer to work as software testers [14].

Our study can be used by employers to calibrate their requests for testers when hiring. In case the job advertisement does not attract testers with exactly the set of skills requested, the employers could either request less or split the responsibilities in other ways. Additionally, our study can serve as a mapping tool for employers to identify skill gaps in the development team. Dependency on key people is reported by Wiklund et al. as the most dominating issue in Agile organisations in need of improvement [83]. Therefore, employers can use our results to examine the spectrum of skills in their development teams and identify how those skills are spread among the team members, compare them with the skills currently in demand in the industry, and either train the existing personnel to gain more skills or hire more skilled testers to the teams.

Our study shows that some of the most in-demand skills focus on the process of test automation. Automation does not mean that the testing happens automatically without the testers having to put work into it; automation is used to improve the effectiveness of testing [84], reduce human errors [85], and improve the reliability of testing, but it is not a human labour replacement. To write, execute, and follow-up on automated tests, a tester must have time and invest a good amount of effort into the process. Therefore, employers need to check whether the testers have the time and skills to develop and maintain a good automation process. This aspect is important, as Karlström et al. report that the main impediments to adopting software test automation are of a managerial and not a technical nature [86]. Also, employers can give more time and the effort needed to maintain the testing infrastructure, as poor-quality infrastructure can be a hindrance for testing and a significant source of loss in software projects [87].

Because there are no clear career paths to follow, software testers can at least gain a better understanding of the skills that are currently important for employers during the hiring process. They can work towards increasing their own skills by taking on tertiary education or certifications or by practicing their skills, especially in the area of test design [88] and test planning [89], test management, performance testing, or acceptance testing. These areas have been confirmed by our study to be important and in-demand in the industry. They can also catch up with their technical side by practicing their programming skills with SQL queries, Java, or C#. They can reflect on how their skills can be transferred to the project management level, especially in the areas of estimation, risk management, planning, reporting, and control.

Our findings show that test managers have to have about as many test-related skills as the testers, and they are asked to have a strong technical profile. The fact that there is a much stronger request for managers to master testing – particularly automated testing – than to manage people, shows the substantial need for qualified leaders who can drive the test automation process. Our previous finding [35] on the demand for independent testers leads to the conclusion that, for the employers in software testing, people management less important compared to the technical qualification of test managers.

The reflection of industrial needs in academic courses and practice has been a long-standing issue. As Garousi et al. state in a recent report [90], even though the software development industry and computer science tertiary education providers are two large communities, the number of joint activities is low. Therefore, an important improvement step towards for the current situation would be for the tertiary education providers to update and expand their curriculum to include the industrial needs, as the results of our current study show. We find the demand for a strong technical profile of the tester particularly interesting. Our results are in line with report [90], highlighting that, with a large margin, the two biggest issues in the industry that are reflected directly in requests for academia are test management and test automation. Therefore, education providers can check whether they offer enough on the side of test automation: from writing to executing tests and setting up automated testing frameworks, preparing data and test oracles, reading logs, following-up on execution results, and so on. It is also important to give the students the chance to practice the continuous integration and deployment of code. In case there are course offerings in software testing, it is important to provide the students the chance to practice high-level test approaches in addition to writing tests. The results of our study show that planning tests, allocating resources, prioritising tests, and designing tests from requirements are highly desirable skills which require practice.

6 Limitations

Obtaining the software tester profile with an industry perspective at hiring is a useful finding in determining what the employers need for the tester role. We made the research data, the protocol we used to create the skill taxonomy and the analysis results publicly available for access and download [44, 49, 50, 91], so that other researchers use it and assert it.

On the other hand, establishing the desired profile of the role at hiring does not mean we can equate it with all skills testers are being asked for throughout their careers. Nor we can equate the testers with the entire testing process; other abilities and critical competencies are the responsibility of other roles, such as developers, managers, customer support, contractors, stakeholders, and the like. Listed below, we have identified other potential threats to the validity of our research.

Generalisation bias: Our analysis stands on observations that were collected from job ads produced by the industry. We cannot generalise our findings to any demand other than industrial or to the demand for the whole job-hiring process. We can also not infer conclusions on the demand for existing employees. We mitigated the limitation using a clear definition of the scope of the research.

Comparison bias: There is no previous study focused on the same parameters as the ones we analysed to make it possible for us to look at trends in the demand. Our study is, therefore, a snapshot of current industrial needs. We mitigated the limitation by finding comparison points with other studies and research and conducted comparisons where possible. We have also used those studies as a base for discussion and for drawing the implications of our study.

Sampling bias: The data were collected at the end 2016 and the beginning 2017, which means that our current analysis results have limited validity, given that it is most likely that industry needs will change over time. An additional limitation for the sample we used in our analysis is that we relied on the search engines' intrinsic logic when we searched for the testing jobs. It could have been that there were additional testing jobs that these engines did not display for the search phrase "software tester". This limitation is palliated by the fact that the search results we obtained are the same as the search results of the job seekers.

7 Conclusion

In this paper, we performed an empirical analysis of the software tester role in which we shaped its profile, as requested by the current needs of the industry. The focus was on making a structured inspection of the skills needed for the role, for which we grouped the demand into corresponding categories in a taxonomy.

The result of our analysis, performed on 400 job ads, shows that software testing is a distinct role that involves a high number of specific competencies that have a strong technical component and a managerial component. Designing of tests is the main industrial demand of software testing skills. The automated testing skill is more asked for than the manual testing skill. It is common among employers to ask the software testers the ability to conduct test planning and implementation. However, it is not common among the employers to ask for skills in bug identification, writing test documentation, generating test data or managing the test closure activities. The software testers are needed to follow-up the reported bugs to resolution, but the employers do not ask the testers for previous experience with specific bug tracking tools. Among the employers, there is a high interest in acceptance testing and a lower interest in asking for integration testing. The software testers are expected to undertake software development tasks and the responsibility for performance testing. Moreover, they are needed to act as quality assurers and to undertake parts of project management and customer support. None the less, there is a strong preference for software testers with a broad technical expertise.

Employers are not likely to be asking for domain-specific knowledge during the hiring process, which denotes the good portability of software testing knowledge across industries. Formal education attainment is moderately important, with relevant bachelor's degrees in computer science or related fields being mostly asked for. One in seven employees is asked to hold a software testing certification. The industry is split in half regarding the demand for previous experience; the half that requires it usually does not demand more than three years of experience.

By exposing the industrial need for various abilities of the software testers, the main contribution of the study is to advance the understanding of the division of labour in software development teams. Empirical studies can be carried-out to analyse the skill demand during the hiring process. Furthermore, the profile of software testers can be broadened with an empirical investigation of skills demand for testers who have already been hired. The current study can be repeated, to observe how the industry shifts its preferences in skills demand.

8 References

1. Byrd, T.A. and D.E. Turner, *An exploratory analysis of the value of the skills of IT personnel: Their relationship to IS infrastructure and competitive advantage.* Decision Sciences, 2001. **32**(1): p. 21-54.
2. Faraj, S. and L. Sproull, *Coordinating expertise in software development teams.* Management science, 2000. **46**(12): p. 1554-1568.
3. Ebert, C. and P. De Neve, *Surviving global software development.* IEEE software, 2001. **18**(2): p. 62-69.
4. Capretz, L.F. and F. Ahmed, *Making sense of software development and personality types.* IT professional, 2010. **12**(1).
5. Jiang, J. and G. Klein, *Software development risks to project effectiveness.* Journal of Systems and Software, 2000. **52**(1): p. 3-10.

6. Fischer, D., *Was Boeing's Compensation Committee Sufficiently Independent in Judging the Business Risk of the 737 Max?* Available at SSRN 3370066, 2019.

7. Flottau, J., S. Broderick, and H. Massy-Beresford, *Airlines struggle to minimize impact of Boeing 737 MAX grounding: carriers search for spare aircraft, replacement capacity; North America and China most affected.* Aviation Week & Space Technology, 2019.

8. Norris, G. and S. Broderick, *MAX issues may force change to long-term fleet-use strategies: pilot input sought on MAX changes; US carriers eye return to flight by mid-August.* Aviation Week & Space Technology, 2019.

9. Edwards, R.C., *Individual Traits and Organizational Incentives: What Makes a" Good" Worker?* Journal of Human Resources, 1976: p. 51-68.

10. Yeatts, D.E. and C. Hyten, *High-performing self-managed work teams: A comparison of theory to practice*. 1998: Sage.

11. Lethbridge, T.C., *What knowledge is important to a software professional?* Computer, 2000. **33**(5): p. 44-50.

12. Lindstrom, L. and R. Jeffries, *Extreme programming and agile software development methodologies.* Information systems management, 2004. **21**(3): p. 41-52.

13. Juristo, N., A.M. Moreno, and W. Strigel, *Guest editors' introduction: Software testing practices in industry.* IEEE software, 2006. **23**(4): p. 19-21.

14. Capretz, L.F., D. Varona, and A. Raza, *Influence of personality types in software tasks choices.* Computers in Human behavior, 2015. **52**: p. 373-378.

15. Stray, V.G., Y. Lindsjørn, and D.I. Sjøberg. *Obstacles to efficient daily meetings in agile development projects: A case study*. in *2013 ACM/IEEE International Symposium on Empirical Software Engineering and Measurement*. 2013. IEEE.

16. Association, I.S. *IEEE 610-1990 - IEEE Standard Computer Dictionary: A Compilation of IEEE Standard Computer Glossaries*. 1990; Available from: https://standards.ieee.org/standard/610-1990.html.

17. Commision, I.E. *ISO/IEC/IEEE 24765:2010 Systems and software engineering - Vocabulary*. 2010; Available from: https://www.iso.org/standard/50518.html.

18. ISO/IEC, *ISO/IEC/IEEE 29119-1: 2013 Software and systems engineering-Software testing-Part 1: Concepts and definitions*. 2013, SwitzerlandInternational Organization for Standardization Geneva.

19. ISTQB, I., *Glossary of Testing Terms.* ISTQB Glossary http://www/. istqb. org/downloads/finish/20/193. html, 2015.

20. Dictionary, C. *Cambridge Dictionary*. Available from: https://dictionary.cambridge.org/.

21. Dictionary, O.E. *Oxford English Dictionary*. Available from: http://www.oed.com/.

22. Dictionary, M.-W. *Merriam-Webster Dictionary*. Available from: https://www.merriam-webster.com/.

23. Dictionary, C. *Collins Dictionary*. Available from: https://www.collinsdictionary.com/.

24. Union, I.T. *International Telecommunication Union*. Available from: https://www.itu.int/en/Pages/default.aspx.

25. Standardization, I.O.f. *International Ogranization for Standardization*. Available from: https://www.iso.org/home.html.

26. Mathur, S. and S. Malik, *Advancements in the V-Model.* International Journal of Computer Applications, 2010. **1**(12).

27. Davidov, M.B., et al., *Method and system for analyzing software test results*. 2010, Google Patents.

28. Ahmed, F., L.F. Capretz, and P. Campbell, *Evaluating the demand for soft skills in software development.* IT Professional, 2012. **14**(1): p. 44-49.

29. Stray, V., T.E. Fægri, and N.B. Moe. *Exploring norms in agile software teams*. in *International Conference on Product-Focused Software Process Improvement*. 2016. Springer.

30. Cohn, M. and D. Ford, *Introducing an agile process to an organization [software development].* Computer, 2003. **36**(6): p. 74-78.

31. Stray, V.G., N.B. Moe, and T. Dingsøyr. *Challenges to teamwork: a multiple case study of two agile teams*. in *International Conference on Agile Software Development*. 2011. Springer.

32. Dromey, R.G., *Software quality—prevention versus cure?* Software Quality Journal, 2003. **11**(3): p. 197-210.

33. Glass, R.L., et al., *Software testing and industry needs.* IEEE Software, 2006. **23**(4): p. 55-57.

34. Bergersen, G.R., D.I. Sjøberg, and T. Dybå, *Construction and validation of an instrument for measuring programming skill.* IEEE Transactions on Software Engineering, 2014. **40**(12): p. 1163-1184.

35. Florea, R. and V. Stray. *Software Tester, We Want to Hire You! an Analysis of the Demand for Soft Skills*. in *International Conference on Agile Software Development*. 2018. Springer.

36. Florea, R. and V. Stray. *A global view on the hard skills and testing tools in software testing*. in *Proceedings of the 14th International Conference on Global Software Engineering*. 2019. IEEE Press.

37. Strode, D.E. and S.L. Huff. *A taxonomy of dependencies in agile software development*. in *ACIS 2012: Location, location, location: Proceedings of the 23rd Australasian Conference on Information Systems 2012*. 2012. ACIS.

38. ISTQB, F., *Foundation Level Syllabus Version 2011.* International Software Testing Qualifications Board, 2011.

39. Indeed.com. *Indeed.com*. Available from: https://no.indeed.com/?r=us.

40. Monster.com. *Monster.com*. Available from: https://www.monster.com/.

41. Glassdoor.com. *Glassdoor.com*. Available from: https://www.glassdoor.com/index.htm.

42. Simplyhired.com. *Simplyhired.com*. Available from: https://www.simplyhired.com/.

43. ISTQB. *ISTQB*. Available from: https://www.istqb.org/.

44. Raluca Florea, V.S. *Analysis of the skill demand for testers*. 2019; Available from: https://doi.org/10.6084/m9.figshare.8294078.v2

45. Ortu, M., et al. *The JIRA repository dataset: Understanding social aspects of software development*. in *Proceedings of the 11th international conference on predictive models and data analytics in software engineering*. 2015. ACM.

46. Jain, A., M. Jain, and S. Dhankar, *A Comparison of RANOREX and QTP Automated Testing Tools and their impact on Software Testing.* IJEMS, 2014. **1**(1): p. 8-12.

47. Avritzer, A. and E.J. Weyuker, *Deriving workloads for performance testing.* Software: Practice and Experience, 1996. **26**(6): p. 613-633.

48. Patel, B., J. Parikh, and R. Shah, *A Review Paper on Comparison of SQL Performance Analyzer Tools: Apache JMeter and HP LoadRunner.* International Journal of Current Engineering and Technology, 2014. **4**(5): p. 3642-3645.

49. Raluca Florea, V.S. *Spearman coefficients, high-level skill analysis*. 2019; Available from: https://doi.org/10.6084/m9.figshare.8299409.v2.

50. Raluca Florea, V.S. *Spearman coefficients, low-level skill analysis*. 2019; Available from: https://doi.org/10.6084/m9.figshare.8299421.v1.

51. Dancey, C.P. and J. Reidy, *Statistics without maths for psychology*. 2007: Pearson Education.

52. Kettunen, V., et al. *A study on agility and testing processes in software organizations*. in *Proceedings of the 19th international symposium on Software testing and analysis*. 2010. ACM.

53. Stevenson, J. and M. Wood, *Recognising object-oriented software design quality: a practitioner-based questionnaire survey.* Software Quality Journal, 2017: p. 1-45.

54. Penov, F.P., et al., *Test execution framework for automated software testing*. 2005, Google Patents.

55. Moe, N.B., et al. *Continuous software testing in a globally distributed project*. in *Global Software Engineering (ICGSE), 2015 IEEE 10th International Conference on*. 2015. IEEE.

56. Collins, E., et al. *An industrial experience on the application of distributed testing in an agile software development environment*. in *Global Software Engineering (ICGSE), 2012 IEEE Seventh International Conference on*. 2012. IEEE.

57. Wiklund, K., et al. *Impediments for Automated Testing--An Empirical Analysis of a User Support Discussion Board*. in *Software Testing, Verification and Validation (ICST), 2014 IEEE Seventh International Conference on*. 2014. IEEE.

58. Bruns, A., A. Kornstadt, and D. Wichmann, *Web application tests with selenium.* IEEE software, 2009. **26**(5).

59. Babinet, E. and R. Ramanathan. *Dependency management in a large agile environment*. in *Agile, 2008. AGILE'08. Conference*. 2008. IEEE.

60. Kassab, M., J.F. DeFranco, and P.A. Laplante, *Software Testing: The State of the Practice.* IEEE Software, 2017. **34**(5): p. 46-52.

61. Laporte, C.Y. and R.V. O'Connor. *Systems and software engineering standards for very small entities: implementation and initial results*. in *Quality of Information and Communications Technology (QUATIC), 2014 9th International Conference on the*. 2014. IEEE.

62. Dias-Neto, A.C., et al., *Toward the characterization of software testing practices in South America: looking at Brazil and Uruguay.* Software Quality Journal, 2017. **25**(4): p. 1145-1183.

63. Forward, A. and T.C. Lethbridge. *The relevance of software documentation, tools and technologies: a survey*. in *Proceedings of the 2002 ACM symposium on Document engineering*. 2002. ACM.

64. Tankard, C., *What the GDPR means for businesses.* Network Security, 2016. **2016**(6): p. 5-8.

65. Aranda, J. and G. Venolia. *The secret life of bugs: Going past the errors and omissions in software repositories*. in *Proceedings of the 31st International Conference on Software Engineering*. 2009. IEEE Computer Society.

66. Rosenkranz, P., et al. *A distributed test system architecture for open-source IoT software*. in *Proceedings of the 2015 Workshop on IoT challenges in Mobile and Industrial Systems*. 2015. ACM.

67. Davis, F.D., *User acceptance of information technology: system characteristics, user perceptions and behavioral impacts.* International journal of man-machine studies, 1993. **38**(3): p. 475-487.

68. Nguyen, H.Q., *Testing applications on the Web: Test planning for Internet-based systems*. 2001: John Wiley & Sons.

69. Lee, J., S. Kang, and D. Lee, *Survey on software testing practices.* IET software, 2012. **6**(3): p. 275-282.

70. Shahamiri, S.R., W.M.N.W. Kadir, and S.Z. Mohd-Hashim. *A comparative study on automated software test oracle methods*. in *Software Engineering Advances, 2009. ICSEA'09. Fourth International Conference on*. 2009. IEEE.

71. Pinkster, I., et al., *Successful test management: an integral approach*. 2006: Springer Science & Business Media.

72. Tian, J., *Software quality engineering: testing, quality assurance, and quantifiable improvement*. 2005: John Wiley & Sons.

73. Nuseibeh, B. and S. Easterbrook. *Requirements engineering: a roadmap*. in *Proceedings of the Conference on the Future of Software Engineering*. 2000. ACM.

74. Stray, V., N.B. Moe, and G.R. Bergersen. *Are Daily Stand-up Meetings Valuable? A Survey of Developers in Software Teams*. in *International Conference on Agile Software Development*. 2017. Springer.

75. Elbaum, S., G. Rothermel, and J. Penix. *Techniques for improving regression testing in continuous integration development environments*. in *Proceedings of the 22nd ACM SIGSOFT International Symposium on Foundations of Software Engineering*. 2014. ACM.

76. Neely, S. and S. Stolt. *Continuous delivery? easy! just change everything (well, maybe it is not that easy)*. in *Agile Conference (AGILE), 2013*. 2013. IEEE.

77. Dietrich, C., et al. *Understanding Linux feature distribution*. in *Proceedings of the 2012 workshop on Modularity in Systems Software*. 2012. ACM.

78. Goodhart, C.A., et al., *An integrated framework for analyzing multiple financial regulations.* International Journal of Central Banking, 2013. **9**(1): p. 109-143.

79. Nirpal, P.B. and K. Kale, *A brief overview of software testing metrics.* International Journal on Computer Science and Engineering, 2011. **3**(1): p. 204-2011.

80. Black, R., *Advanced Software Testing–Guide to the ISTQB Advanced Certification Vol. 1 and 2.* ISBN-13, 2009: p. 978-1.

81. Garousi, V. and J. Zhi, *A survey of software testing practices in Canada.* Journal of Systems and Software, 2013. **86**(5): p. 1354-1376.

82. Poon, P.-L., et al., *Contributions of tester experience and a checklist guideline to the identification of categories and choices for software testing.* Software Quality Journal, 2011. **19**(1): p. 141-163.

83. Wiklund, K., et al. *Impediments in agile software development: An empirical investigation*. in *International Conference on Product Focused Software Process Improvement*. 2013. Springer.

84. Mahmud, J., et al., *Design and industrial evaluation of a tool supporting semi-automated website testing.* Software Testing, Verification and Reliability, 2014. **24**(1): p. 61-82.

85. Rafi, D.M., et al. *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*. 2012. IEEE Press.

86. Karlström, D., P. Runeson, and S. Nordén, *A minimal test practice framework for emerging software organizations.* Software Testing, Verification and Reliability, 2005. **15**(3): p. 145-166.

87. Kasurinen, J., O. Taipale, and K. Smolander, *Software test automation in practice: empirical observations.* Advances in Software Engineering, 2010. **2010**.

88. Martin, D., et al. *'Good'organisational reasons for'Bad'software testing: An ethnographic study of testing in a small software company*. in *Proceedings of the 29th international conference on Software Engineering*. 2007. IEEE Computer Society.

89. Toroi, T., A. Raninen, and L. Vaatainen. *Identifying process improvement targets in test processes: a case study*. in *Software Maintenance (ICSM), 2013 29th IEEE International Conference on*. 2013. IEEE.

90. Garousi, V., et al. *What industry wants from academia in software testing?: Hearing practitioners' opinions*. in *Proceedings of the 21st International Conference on Evaluation and Assessment in Software Engineering*. 2017. ACM.

91. Raluca Florea, V.S. *Processed data, software testing jobs*. 2019; Available from: https://doi.org/10.6084/m9.figshare.8294054.v1