TÜBİTAK

Research Article

A preliminary survey on software testing practices in Khyber PakhtunKhwa region of Pakistan

Bushra LATIF , **Tauseef RANA***

Department of Computer Software Engineering, National University of Sciences and Technology, Islamabad, Pakistan

Abstract: Conducted to ensure the quality of software products, the software testing process has a great significance in the software development and is the vital step of the verification and validation process. For conforming a software feature to the end user requirements, organizations rely on extensive testing procedures. Despite being the key factor, many of the software development industries/companies do not define/follow a systematic testing process. In this paper, we analyze/learn from the conducted surveys in the past and formulate a questionnaire for a survey in the northern region of Pakistan. To the best of our knowledge, no such survey has ever been conducted in Pakistan. Thus, our work can be used as a baseline for such surveys in other regions or for a survey at the country level. This survey focuses on various aspects of testing methodologies and techniques. With the analyses of the survey findings, we conclude that a testing "maturity" model (based on the cultural/organizational/political values/constraints) should be devised for the improvement of current testing practices to ensure the production of quality software. The survey is regional but the results and knowledge obtained are in line with the related surveys conducted in other regions of the globe.

Key words: Empirical software testing, Software testing practices, software quality, Khyber PakhtunKhwa Pakistan, software development verification and validation process

1. Introduction

Software testing is one of the key processes used to ensure software quality by software developers. Software development companies use various software testing approaches to improve the quality of software products. To ensure effective software testing in the software development life cycle (SDLC), different testing approaches, techniques, and methodologies supported by semi-/fully automated software tools are used. Various software testing strategies should be planned to apply at different levels (e.g., for unit testing, integration testing, and system testing) by using different techniques (e.g., functional or structural or scenario-based testing) for achieving better software testing.

Software testing has always been a concern for the software development industry all over the world. An international body of knowledge named International Software Testing Qualifications Board (ISTQB)[1] carried out a worldwide survey (ISTQB 2015-2016) on software testing practices adopted around the globe. It has received responses from more than 3200 respondents from 89 countries with well-balanced geographical locations. The results of this survey reveal that (i) 81% of the respondents reported to use a defect tracking

---

*Correspondence: tauseefrana@mcs.edu.pk/gmail.com

[1]ITQB(2015-2016).Worldwide Software Testing Practices Report 2015-2016 [online].Website http://www.istqb.org/ [accessed 15 Sept 2017]

tool, (ii) 72% of respondents stated that at least 20% of the test cases are automated and 58% reported that test automation is an area offering improvement in their organizations, (iii) the most common testing stop criteria are test coverage (73%) and the scheduled deadline (46%), and (iv) the most adopted testing techniques are use case testing (70%), exploratory testing (66%), and checklist-based testing (54%).

To the best of our knowledge, no survey on software testing practices has been conducted so far in Pakistan. In this paper, we present the first ever such survey and its results. It is important to address the government efforts for the promotion of IT sector in Pakistan. This survey focuses on the study of testing practices adopted in an underdeveloped region of Pakistan. From the related surveys conducted in the past, we formulate a questionnaire for our survey that covers various aspects of testing methodologies and techniques. The quality of a software application (measured by many attributes such as performance, reliability, and efficiency) is checked and ensured by executing a carefully planned testing phase. For this testing phase, the main focus of our survey is to analyze different testing practices and to suggest ways to streamline these testing practices for achieving better software products.

There are different software development companies in Pakistan operating under the umbrella of a public sector platform for the promotion of the IT sector, named Pakistan Software Export Board (PSEB). Khyber PakhtunKhwa (KPK) is a province in Pakistan and is located in the north-west region. Compared to other advanced regions, KPK's IT sector is still in the developing phase. Working for the promotion of the software industry, KPK Information Technology Board (KPITB) is a provincial organization recognized under the Law Amendments made in 2011. A quality software product, meeting the user's specified requirements, is the desire/intention for the newly emerging industry in such developing regions (under PSEB and KPTIB). This survey was conducted through KPITB to find out the software testing methodologies and techniques that can improve the software quality in KPK.

In this survey, we target firms operating under PSEB or KPITB or acting independently, to identify the deficiencies in their testing procedures. The survey is regional but the results gained are in line with the related surveys conducted in other regions of the globe [1–3]. Surveys, similar in nature, were also carried out limited to some geographical regions; for example, a survey for New Zealand [1], a survey for Canada [2], and a survey for Turkey [3]. In general, the results of these surveys reveal that software development organizations do not exactly cover the major aspects of testing practices which are considered vital for qualified/tested software. With few exceptions, software testing techniques are still not being practiced at large [4].

Focusing on the research questions (mentioned in Section 3.1), an online Google survey form was created (shown in Appendix A2). This form was shared with 150 organizations and the results summarized in this paper are based on responses of the 50% of the contacted organizations. Approximately, all the organizations that responded to our survey are associated with PSEB; around 40 are registered with KPITB and the others either act independently or are in the process of registration with PSEB or KPITB. Companies registered with PSEB and KPITB have to follow the imposed standards by these registration bodies; however, the others follow different ways. Hence, the summarized results are a combination of various aspects of IT practices adopted in firms operating under different umbrellas. Results show identifications of techniques, tools, methodologies, and adherence to international and inland testing standards in KPK. The results also help in the recognition of importance of testing practices and improvements of these practices in the academia. The results of this survey will be a benchmark for future studies and solutions in the field of software testing. Moreover, it would assist Pakistan software industries to improve the software quality and to identify the best testing strategies.

The rest of the paper is organized into six sections. Section 2 covers the necessary background of the software quality subject and the related work. Section 3 explains the proposed methodology of the survey for KPK region in Pakistan while Section 4 elaborates empirical results of the survey followed by discussion in Section 5. Finally, the paper is concluded in Section 6.

2. Related work and objectives of our survey

For an IT firm/organization, software testing is a mandatory process/task for the validation of the system under development. Many leading IT firms and software development organizations face many intricate issues with respect to the technological swings [5] which includes the technologies developed by them or used by them for the development or for the execution of the developed systems. Despite being defined in many different ways imprecisely, without any doubt, quality is the most desired aspect of any software by its stakeholders.

Software testing is an essential part of developing quality software. In SDLC, some of the software testing techniques and methodologies presented in the literature are: (i) static vs. dynamic testing, (ii) box (testing) approaches, (iii) testing at different levels (integration testing, unit testing, and system testing), (iv) testing types (installation testing, compatibility testing, smoke testing, regression testing, and acceptance testing), (v) alpha testing, and (vi) beta testing. In SDLC, some test processes are: (i) agile or extreme testing, (ii) top-down or bottom-up testing, (iii) testing in the traditional water fall model, and (iv) automated testing by using tools [3].

There are different types of software systems. Depending on its function type (e.g., noncritical, safety-critical, and mission critical) and working environment, a system is tested differently according to its context [25]. In short, testing is basically dependent on the nature of a system [6]. Software testing is a costly process; however, through automated tools the resources (testing duration and man-effort) can be saved with much improved results [5, 7, 8]. Many different surveys (with specific focus) have been conducted on the subject of software testing practices. In the rest of this section, we briefly mention prominent examples of such surveys.

A survey of eleven Swedish software development companies shows the common use of incremental software testing and day-to-day build approach [9]. For code reusability, Torkar and Mankefors reveals that testing practices are not used in code reuse [10]. Another survey reveals that the lack of developer expertise is an obvious obstacle in software testing methodologies [4]. In Finland, 18 experts were invited from different organizations and universities to analyze the problems in software testing research [11]. A framework developed by Sung [1] encompasses important parts of software testing by incorporating the current practices of testing. This framework was used in a survey revealing that current trends of testing are lacking many major aspects of software testing. In 2006, twelve European organizations' test managers were interviewed by some researchers regarding development of software testing practices. The researchers proposed a solution to focus on program metrics for economic-based quality of a product [12]. Moreover, the dire need of strong integration between research and industry was also highlighted [13]. For regression testing, in a survey it was found that most of the participants were of the view that the regression testing can be done in different context with prerequisites, but automated testing and conducive test environment have substantial benefits [14].

The researchers of a quantitative survey categorized the participants of the survey in four groups with respect to the safety-critical element, distribution of development, the agility domain, and the application domain [15]. In another survey, researchers found that Canadian software development organizations are giving more attention to the significance of software testing practices to obtain qualified software and more such surveys

are needed to be conducted to measure the development of software testing practices [2]. In a survey conducted in Turkey [3], several aspects were covered; for example, type of software testing practices used and the ratio of software testers to developers in Turkish organization

There are many surveys conducted regarding software testing but each survey has a different point of focus and context. Table 1 shows the conducted surveys in the literature with their respective objectives (further details are shown in Appendix A1). These surveys fall into different categories based on the scope. It can be concluded from Table 1 that only four surveys conducted in Australia, New Zealand, Canada, and Turkey have the similar scope of work with the survey presented in this paper. In these surveys, authors mainly focus on the study of testing practices adopted in their respective regions.

Table 1. Surveys objectives comparison

Categories	Paper Reference	Region	Objectives
Development	Gelprin & Hetzel(1988)	Washington USA	To describe the major(test) process model and explain variation related to testing growth.
Development	Torkar & Mankefors(2003)	Sweden	To what extent software testing had been used in the reusability of software components
Testing Practices	Ng et al(2004)	Australia	To conclude the type, techniques, tools, metrics and standards of testing used by organizations in Australia
Research	Taipal et al(2005)	Finland	To analyze the problems in the software testing research
Testing Practices	Sung & Paynter(2006)	New Zealand	Comparative study of software testing practices with a defined framework by Sung
Academic and Research	Martin & David(2007)	UK	To describe the difference between software testing in the field of academic research and its implementation in the industry.
Development	Ambler (2009)	Not Reported	To observe the ratio of TDD related techniques and comparison of agile practices by the developers.
Development	Engstrm,Emelie(2010)	Sweden	To identify the gaps between the state of the art and the present procedure followed for regression testing practices.
Testing Practices	Garousi & Zhi(2013)	Canada	To describe the testing practices followed in canada.
Testing Practices	Garousi et al(2015)	Turkey	To describe the testing practices and to deliver a view of the latest tools, methods & metrics used and the challenges confronted by testers

Keeping in mind the surveys conducted in Australia, New Zealand, Canada, and Turkey, Table 2 shows the comparative analysis of the objectives covered in these surveys and our intended survey's objective as shown in the last column. This shows that our survey is more comprehensive in number of issues covered compared to the existing surveys.

3. Our survey methodology

In this section, we devise an improvised survey, based on the surveys conducted (to the best of our knowledge) in the literature (described in Section 2). The goal of our survey is to analyze the software testing practices of IT development sector.

Table 2. Survey questions mapping to research questions

County	Australia	Canada	New zealand	Turkey	Pakistan
Testing Techniques	yes	yes	yes	yes	yes
Automated Tools	Yes	Yes	Yes	Yes	Yes
Testing Metrics	Yes	No	Yes	Yes	Yes
Testing Standards	Yes	Yes	Yes	Yes	Yes
Training/Education	Yes	Yes	Yes	No	Yes
Testing Documentation	Yes	Yes	Yes	No	Yes
Barriers to Adopt Testing	Yes	Yes	Yes	No	Yes
Independent software testing team	Yes	Yes	Yes	No	Yes
When to terminate testing activities	Yes	Yes	Yes	yes	No

3.1. Research questions

To fix our survey's goals, the idea of devising a set of research questions is taken from [16]. Questions are divided into 4 sections in the questionnaire (shown in Appendix A2). Section three is the key part of questionnaire (shown as Question No. 5-21 in the survey) which focuses on the organization's testing practices, tools, techniques, metrics, any training for software testing is conducted and the barriers that prevent organizations from conducting testing training. The research questions (1, 2, and 5) also focus on the same points on which section three of the questionnaire focuses. Mapping of research question to survey questionnaire are shown in Table. There is no mapping for screening question (Question No. 1–3, 15, and 25).

Table 3. Survey questions mapping to research questions.

| Research questions	Survey questions
RQ1: What are the adopted testing techniques by software development organization in KPK, Pakistan	6–9,13
RQ2: What are the organization policies to manage the test team and the testing process?	4,12,16–21
RQ3: What techniques are followed to mitigate risks?	9,22
RQ4: What is the importance of software testing in the software development life cycle for an organization?	5,11,23,24
RQ5: What are the obstacles to adopting testing practices?	10,14

3.2. Survey design

In the light of the information gathered from existing surveys (shown in Table S), the survey is designed to gather information about testing training in organizations, advantages of different automated testing tools, and risk mitigation techniques and methodologies. However, the fundamental objective was to investigate whether software testing was considered a valuable part of development or not. Which testing tools, levels, approaches and standards are mostly used in the organizations of software development? A survey questionnaire is devised (as four sections shown in Appenidx A2) and directed to firms through PSEB and KPITB.

The first section includes the profile of the participant on behalf of an organization (shown as Question No. 1 in the form). The second section of the questionnaire is about the organizational information (shown as Question No. 2–4 in the form). The third section is the key part of the questionnaire (shown as Question No. 5–21 in the form) which focuses on the organization's testing-related information. The fourth section attempts to confirm the importance of software testing practices (shown as Question No. 22-25 in the form), risk mitigation factors, and testing methods mostly used in KPK software industry.

3.3. The survey method and flow

Figure 1 reflects the flow of survey execution. The software development companies registered with KPITB and PSEB in the region of KPK Pakistan represent the targeted population of the survey. As per our knowledge gained from PSEB, more than 130 companies are registered with PSEB. An email sharing list concerned with software testing practitioners was used to maintain the participants in the region. The practitioners of software
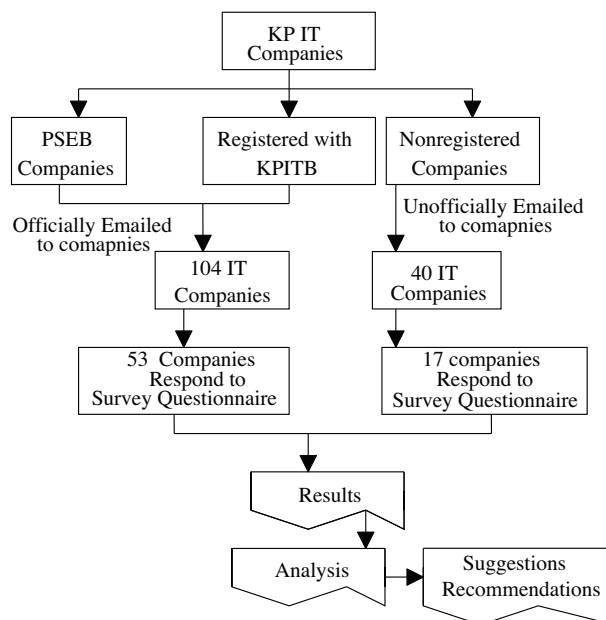
Figure 1. Flow of survey execution.

development companies were sampled by suitability. By suitability we mean the relevance of a person to his/her particular job, so the practitioners of software development companies were sampled to participate in the survey as they were well aware of the testing procedures of the companies adopted for testing. To get fair results, the request/invitation to participate in the survey was sent by email/phone to the companies registered with PSEB and KPITB. Furthermore, some other companies (whose registration with the PSEB and KPITB are in process) were also approached. Moreover, some privately run organizations were also approached to participate in the survey.

Web-based survey was chosen for conducting the survey as internet being in common access nowadays. The survey was designed by the Google Form facility and hosted by the Google Derive. The questionnaire form was made available to the participants for a month, March 2018. For pursuance, emails were sent with a link to the participants and phone calls were made. The questionnaire form includes 25 questions divided into

four sections. The survey questionnaire was sent to 144 software development organizations in different cities (Peshawar, Swat, Mardan, Abbottabad, Swabi, Hangu, and Bannu) of the KPK region. The list of the software development companies in KPK was obtained from KPITB. During the scrutiny process, it has been observed that 7 companies were closed, and healthy number of companies were registered with KPITB and PSEB as well. To get maximum response from the contacted companies, an official email through KPITB and PSEB was also initiated to make the results more clear and real. Confidentiality was maintained for participants' responses. At the end, after the preprocessing phase, response data from 70 companies across KPK were obtained.

4. Survey results and findings

For this survey, a total of 144 organizations were emailed and we received 70 responses from different cities of KPK. These responses are mostly from the software development companies from the commercial sector; hence, main focus of our survey is in the testing practices followed in the commercial organizations. In the rest of this section, the findings of the survey with respect to its different structural sections are reported.

4.1. Survey participants and organizational information

From the survey respondents' data, it is learned that most of the individuals have a university level degree and 4-5 years of experience in the field of software testing. A good experience of such professionals with appropriate university degree may lead the IT firms in KPK to establish their own software quality assurance team for software testing, thus avoiding hiring of external testers and reducing such expenses.

In our focused region of Pakistan, most of the organizations were small with average working manpower of 20–30 employees; however, some were large with more than 100 employees. To conclude sector-wise information, we try to categorize the organizations into commercial, government, and private sectors. It is found that almost $45(\frac{45}{70} \times 100 \cong 64\%)$ companies are commercial, 21 ($\frac{21}{70} \times 100 \cong 30\%$) are government organizations, and 5 ($\frac{5}{70} \times 100 \cong 7\%$) respondents fall in the private sector categories. It is revealed from the survey that most of the KPK IT firms are operating in the commercial sector category. Moreover, in this section, the responsible individual for software testing is questioned to get the testing status of an organization. Around 12 out of 70 ($\frac{12}{70} \times 100 \cong 17\%$) companies responded that developers perform testing activities in parallel to other opposite nature responsibilities. Only one ($\frac{1}{70} \times 100 \cong 1.4\%$) company claims to have a dedicated software testing team.

4.2. Software testing practices

The key section regarding software testing practices of the survey form (comprised of 17 questions) reveals different aspects of the IT industry in KPK. This important part (of the form) is described in this subsection.

4.2.1. Training

From the responses of Question No. 5 and Question No. 11, Figure 2 represents that out of 70 organizations, more companies respond that there is no testing training sessions in their companies and some organizations conduct training of their staff. The familiarity of the testers with the automated testing tools is beneficial for the quality assurance process in an organization. Around 40 organizations ($\frac{40}{70} \times 100 \cong 58\%$) reported that no training sessions for any specific automated tools were ever conducted by them. Software testing practices in KPK seem to be restricted mainly due to the lack of resources and the experienced personnel.

4.2.2. Structural/functional testing

Figure 3 shows the response to Question No. 6 (regarding the black/white/grey box testing approaches); most of the organizations depend on the white box technique as 40 out of 70 ($\frac{40}{70} \times 100 \cong 57.1\%$) companies use the white box approach. In the white box testing the understanding of the code is paramount. Therefore, developers act as tester to carry out the white box testing by themselves. Doing this avoids hiring external testers, which results in reducing the cost.
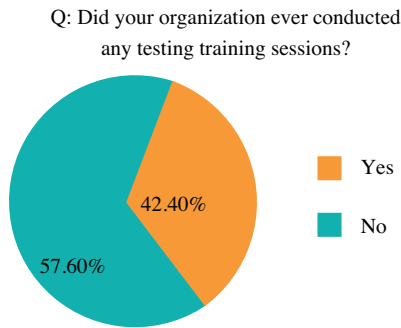
Figure 2. Testing training.
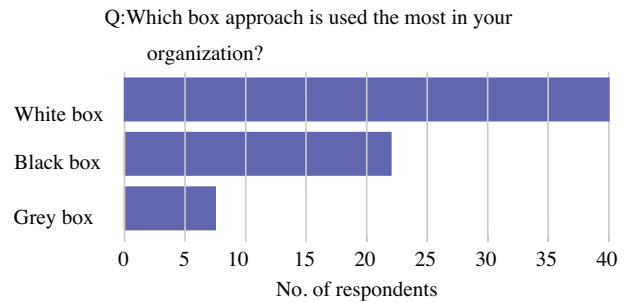
Figure 3. Box testing approaches.

From the responses of Question No. 7 shown in Figure 4 (about the use of different testing levels which are unit, integration, system, and acceptance), unit testing dominates and it is claimed that it enables a developer to fix defects sooner. Hence, fixing bugs faster saves time to deliver product and reduce expenditures. At the unit level, single units of software are tested to confirm that each unit of the software performs as intended. At the integration level, individual units are collectively tested as an assembly (group) to uncover errors in the interaction between collective units. At the system level, the whole/joined system is tested to gauge the system's compliance with the identified/specified requirements. At the acceptance level, acceptability of the system is tested to evaluate its submission for delivery.

4.2.3. Techniques for test cases

In order to validate a system according to the given requirements/specification, test cases are created. For the preparation of test cases, a number of testing techniques can be used; for example, boundary value analysis, equivalence partitioning, use of control flow graphs, and out of range technique [2]. From the respondents' data for Question No. 8, Figure 5 reveals the use of testing techniques for test case creation.
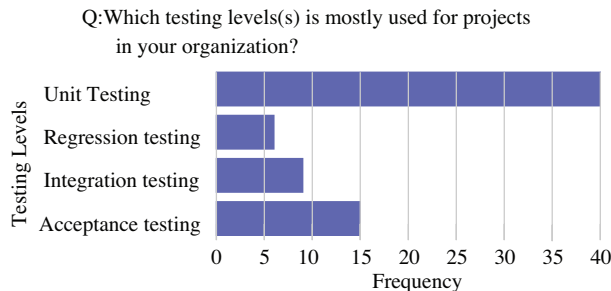

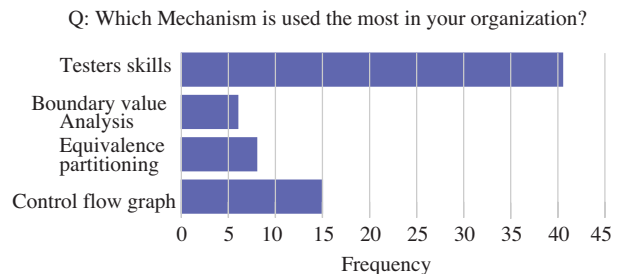**Figure 4**. Testing Levels

Figure 5. Mechanism for test cases.

4.2.4. Software metrics

In software testing, a number of metrics are used to measure/observe testing activities [17]. Testing metrics are divided into two categories: base and calculated. In the base category, the data is gathered during the test case preparation and execution. In the calculated category, metrics are derived from the data gathered in the base metric.

For the preparation of Question No. 9 of the survey, the metrics are taken from ToTheNew[2]. The definition of these metrics are shown in Figure 6. Test case productive preparation falls in the base category; this is used to calculate the number of test cases created and the effort made in the preparation of test cases. In test case coverage metrics, number of lines within the code and number of lines executed by tests are used. Responses of Question No. 9 are shown in Figure 6. Learned from the survey data (shown in Figure 6), commonly used testing metrics include test case productive preparations used by 26 firms and test case coverage opted by 23 respondents.
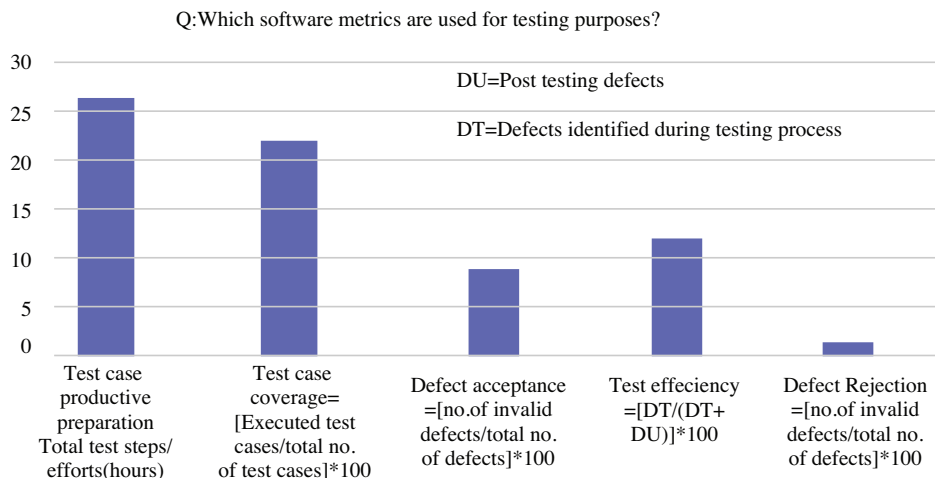

Figure 6. Use of Software Metrics

4.3. Barriers for testing methodologies

Being in a comparatively underdeveloped region of a developing country, there can be various barriers in adopting testing techniques/tools faced by the workers in an organization. To collect the reasons of these hurdles, Question No. 10 and Question No. 14 are included in the survey. From the responses of Question No. 10, Figure 7 shows the key barriers/hurdles faced by different firms participating in our survey. In the survey, shown in Figure 7, 29 respondents ($\frac{29}{70} \times 100 \cong 41.4\%$) consider the time (scheduled delivery of product, time frame for the development cycle or time constraints in the utilization of testing tools) as the key barrier in the way of proper software testing. The level of difficulty or the absence of proper user guidance for the use of a tool is another hindrance. Cost and difficulty of use of tools are other major hurdles. In connection to the cost, Question No. 21 is posed directly. In KPK, among the software development companies, there is low trend of proper budget allocation to software testing activities.

---

[2]ToTheNew (2016).Software Testing Metrics [online].Website http://www.tothenew.com/blog/software-testing-metrics/[accessed 12 june 2019]

From the respondents (shown in Figure 8), $46(\frac{46}{70} \times 100 \cong 65.7\%)$ organizations do not allocate any budget for the testing process. Besides time and cost, another barrier concluded from the survey is the lack of expert tester available in the organizations.

Q:What are the barriers for your organization to adopt the testing tools?

Figure 7. Barriers for testing methodologies

Q: Does your organization allocate any budget for software testing process?
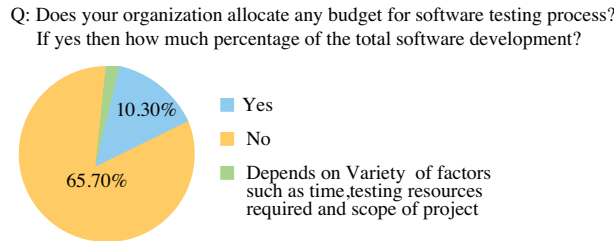If yes then how much percentage of the total software development?

Figure 8. Budget Allocation

4.4. Use of automated testing tools/standards

We intended in our survey (by Question No. 13 as shown by the responses in Figure 9) to measure the responses on the use of automated testing tools or automation of software testing practices. From the respondents, 17 ($\frac{17}{70} \times 100 \cong 24.2\%$) firms use tools developed in-house for test automation while 25 ($\frac{25}{70} \times 100 \cong 35.7\%$) organizations use open-source tools. However, 9 ($\frac{9}{70} \times 100 \cong 12.8\%$) respondents use a combination of both open-source and in-house–developed tools. Testing can be done manually but there is a chance of human errors while automated testing tools reduce such risks of human errors. Due to the human error factor, the quality of product can be degraded. Using automated testing tools, the quality of a product can be enhanced.

With respect to the use of testing standards (the survey Question No. 19), 54 firms ($\frac{54}{70} \times 100 \cong 77.1\%$) shown in Figure 10 claim that they do not follow any software quality standards like ISO/IEEE/IEC. While 6 ($\frac{6}{70} \times 100 \cong 8.5\%$) organizations use ISO 9001 standards during their testing practices. It is duly highlighted that the use of quality standards must always be incorporated in the quality assurance policy adopted by any firm. Such standards are considered mandatory for quality improvement of products.
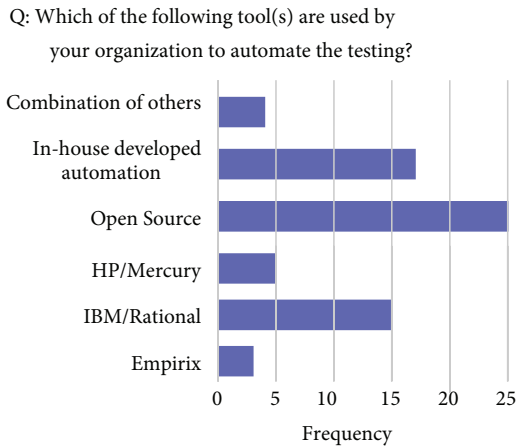
Q: Which of the following tool(s) are used by your organization to automate the testing?

Figure 9. Use of Automated Testing tools

Q:Does any international software quality standard is followed in your organization? If yes then specify.
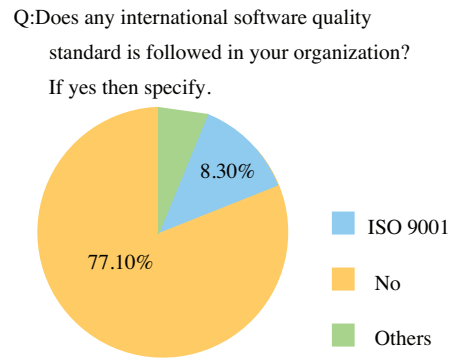

Figure 10. Use of Testing Standards

4.5. Importance of software testing

In software project management, identification of risks and mitigation of these risks is covered under the risk management activity [25]. Risk-based testing is a well-established concept [22]. In our survey, incompetence of a software to meet the specifications is referred to as the risk. Many features of the software can only be checked by executing the software; this is the testing phase. There are many different testing techniques used to mitigate the aforementioned risks [3, 25]. From the survey data (shown in Figure 11), most of the organizations follow testing level and box approaches for risk mitigation. It is concluded that 29 ($\frac{29}{70} \times 100 \cong 41.4\%$) organizations adopt the testing level in risk mitigation. While 22 ($\frac{22}{70} \times 100 \cong 31.4\%$) firms report that box approach is better in risk mitigation.

Last section directly refers to the importance of software testing practices that show how risks can be mitigated through proper implementation of the software testing practices and how much the metrics help in the efficiency and the effectiveness of the testing process. Figure 12 represents that 25%–50% of testing phase effectiveness is dependent on software testing metrics.
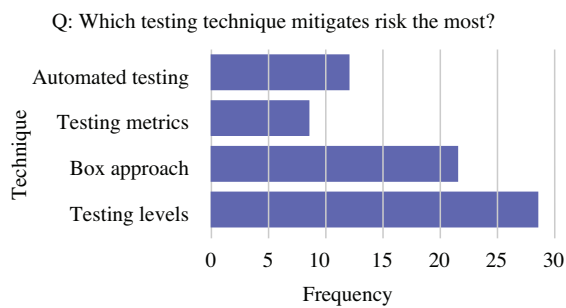
Q: Which testing technique mitigates risk the most?

Figure 11. Risk Mitigation

Q: How much software testing metrices are effective in the software effeciency?
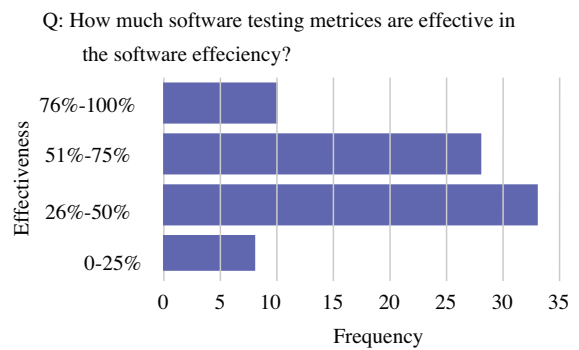

Figure 12. Effectiveness of software testing metrics

5. Discussions and the way forward

From the survey form, Questions No. 12, 15–18 are not covered in Section 4. For Question No. 15–18, deterministic information about the testing procedures through the responding organizations are provided; that is why they are not included in Section 4 but discussed in this section.

The prime objective of this survey is to explore the status of software testing practices in KPK, Pakistan. Based on the received responses it is learned that the software companies are small in size (approx. 5-40 employees). These companies are striving to survive in the competitive market of development by undertaking national/international software projects. For better business progress, these companies must produce high-quality software. Despite the governmental efforts (in the forms of PSEB and KPITB bodies to guide), the survey results show weaknesses in the practices of the surveyed organizations. In order to achieve high-quality software, the companies should have more updated technological infrastructure and resources available; however, as the IT sector is still in the development phase in KPK, the software companies are smaller in size and have lack of resources to properly manage testing policies. A very low number (1.7%) of respondents claim that they have dedicated software testing teams and project leaders who are mainly responsible for software testing activities, which reveals that there is lack of organizational structure in the KPK software industry and this problem may lead to chaotic environment.

For the KPK-based software industry, among different software quality standards ISO/IEEE/IEC, there is no proper use of a standard and most of the companies are not even aware of these standards. For a project/product to be a quality software, it has to conform to the user requirements. In order to fulfill users' expectations and requirements, test cases are fundamental to check the operational capability of a software in the operational environment. In KPK, for more than 60% of the software companies, test cases for end products are devised on the basis of testers' own experiences. No standardized test cases are formulated keeping predicted inputs/outputs to the system. Recorded test case results can be reused as reference in future projects and can trace user requirement. In this regard, for better decision making, the techniques of machine learning can be used; moreover, the usage of artificial intelligence in this regard would be beneficiary.

There is no defined criteria for when to stop testing a product. For better resource management and to cut unnecessary prolonging the testing process, the criteria to stop testing should be predefined. Cost, lack of expertise, improper budget allocation, and available time for a project are the main constraints in adopting a better software testing practice. Around 43 out 70 respondents were of the view that there is no proper allocated budget for testing phase in the development cycle, whereas few say that they utilize 20%–30% of the total product/project budget for the testing phase. Lack of regular professional training hampers the end product quality in a serious way. This indicates that there might be several software testers in a firm who are not educated/trained for the testing work.

With the analysis of the survey findings, we realize that a "maturity" model (based on the cultural/organizational/political values/constraints) should be devised for the improvement of current testing practices to ensure quality software. This model should be in line with the recommendations of the testing maturity model (TMM) [21]. The self-descriptive structure of TMM is shown in Figure 13.

To devise such a software testing model, various factors should be kept in mind; for example, a separate testing team needs to be organized with each individual having different skill levels, i.e. knowledge, capability, diverse approaches, and different prospects/interests levels. When these levels do not match with the requisite product, the organizational/political issues may arise. The team must work jointly and follow the organizational
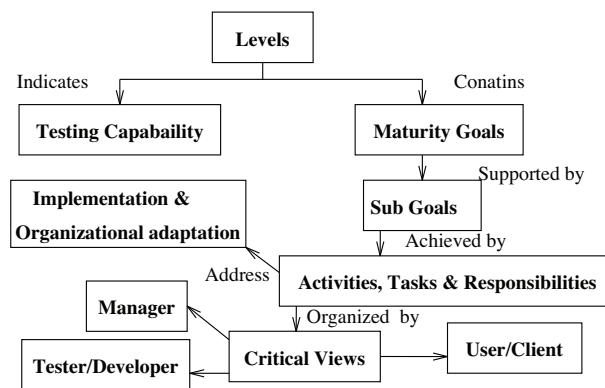
Figure 13. Structure of TMM

norms/procedures to distribute the stipulated portion of work within the planned time. All of these require concerted efforts for test management, which is often performed by an individual being in the role of team leader. However, if these defined standards are not followed or there is mismanagement of all the activities then this would directly lead to organizational issues.

Sometimes, adopted under political/cultural constraints, a testing tool may not be appropriate for the intended work. The purpose of the suggested model is to bring the testing activities of the development industries of KPK region streamlined and to overcome the gaps/grey areas identified from the survey in the testing procedures by implementing the model.

This research has revealed several aspects missing from the software companies of KPK. Adaptation of these influences to advance quality of the software artifact is of main concern. However, the skills of an organization to implement these practices is still undiscovered and can be researched in future studies.

6. Conclusion and future advancements

This paper presented a survey conducted to analyze the level of software testing practices in software development firms in the KPK region of Pakistan. The survey was conducted in February 2018. IT sector in KPK region is in the developing phase, yet 70 organizations have responded and the results were compiled.

Concluding with the survey, we have come to know that KPK IT firms are of small size and inexperienced in using a standard testing procedure. Hence, appropriate software testing practices are mostly ignored due to the lack of expertise and multitasking of the men power. Main barriers (as shown in Section 4.3) to follow standards in the testing practices and noncompliance to the defined procedures lead to a weak link between software development cycle and testing practice. In view of Section 3 in the survey (as shown in Appendix A2), the key purpose of the survey was to explore the hurdles in practicing the testing techniques in software development organizations. Furthermore, it is hoped that the results of this survey will be used for improving the testing practices Moreover, this survey can be expanded to see the state of testing practices in other regions of Pakistan or to study the state at the country level.

Knowing the weaknesses in the current testing practices motivates us to explore the possibilities of studying existing testing models (e.g., a reference model is shown in Figure 13) for its possible usage with/without modifications for the resolution of the identified weak areas through this survey and device a "maturity" model for the improvement of current testing practices to ensure quality software. The purpose of the model is to make the testing activities of the development industries of KPK region streamlined and to overcome the gaps/grey

area identified from the survey in the testing procedures by implementing the devised model [18–21]. In order to address the identified weaknesses in this survey, we need to pursue the development of the aforesaid "maturity" model by study and analysis of existing models (e.g., test improvement model (TIM) [18], test process improvement model (TPI) [18, 24], critical testing process (CTP) [18, 23], testing assessment program (TAP) [21], and testability support model (TSM) [18]) in a future study.

References

[1] Sung PW, Paynter J. Software testing practices in New Zealand. In: National Advisory Committee on Computing Qualifications Conference; New Zealand; 2006. pp. 273-282.

[2] Garousi V, Zhi J. A survey of software testing practices in Canada. Journal of Systems and Software 2013; 86(5): 1354-1376. doi: 10.1016/j.jss.2012.12.051

[3] Garousi V, Coşkunçay A, Betin-Can A, Demirörs O. A survey of software engineering practices in Turkey. Journal of Systems and Software 2015; 108: 148-177. doi: 10.1016/j.jss.2015.06.036

[4] Ng SP, Murnane T, Reed K, Grant D, Chen TY. A preliminary survey on software testing practices in Australia. In: IEEE Software Engineering Conference; Melbourne, Victoria, Australia; 2004. pp. 116-125.

[5] Wissink T, Amaro C. Successful test automation for software maintenance. In: IEEE International Conference of Software Maintainance; Philadelphia, PA, USA; 2006. pp. 265-266.

[6] Parnin C, Orso A. Are automated debugging techniques actually helping programmers? In: ACM International Symposium on Software Testing and Analysis; New York, NY, USA; 2011. pp. 199-209

[7] Du Bousquet L, Zuanon N. An overview of Lutess a specification-based tool for testing synchronous software. In: IEEE Automated Software Engineering Conference; Cocoa Beach, FL, USA; 1999. pp. 208-215.

[8] Karhu K, Repo T, Taipale O, Smolander K. Empirical observations on software testing automation. In: IEEE Software Testing Verification and Validation Conference; Denver, CO, USA; 2009. pp. 201-209.

[9] Runeson P, Andersson C, Höst M. Test processes in software product evolution—a qualitative survey on the state of practice. Journal of Software Maintenance and Evolution Research and Practice 2003. 15(1): 41-59. doi: 10.1002/smr.26.

[10] Torkar R, Mankefors S. A survey on testing and reuse. In: IEEE International Conference on Software Science Technology and Engineering; Herzlia, Israel; 2003. pp. 164-173.

[11] Taipale O, Smolander K, Kälviäinen H. Finding and ranking research directions for software testing. In: Software Process Improvement European Conference; Budapest, Hungary; 2005. pp. 39-48.

[12] Grindal M, Offutt J, Mellin J. On the testing maturity of software producing organizations. In: IEEE Testing Academic and Industrial Conference-Practice And Research Techniques; Windsor, UK; 2006. pp. 171-180.

[13] Martin D, Rooksby J, Rouncefield M, Sommerville I. 'Good'organisational reasons for'Bad'software testing: An ethnographic study of testing in a small software company. In: IEEE Proceedings of the 29th international conference on Software Engineering; MN, USA; 2007. pp. 602-611.

[14] Engström E, Runeson P. A qualitative survey of regression testing practices. In: International Conference on Product Focused Software Process Improvement; Limerick, Ireland; 2010. pp. 3-16.

[15] Causevic A, Sundmark D, Punnekkat S. An industrial survey on contemporary aspects of software testing. In: IEEE International conference on Software Testing, Verification and Validation (ICST); Washington, DC, USA; 2010. pp. 393-401.

[16] Nanthaamornphong A, Carver JC. Test-Driven Development in scientific software: a survey. Software Quality Journal 2017. 25(2): 343-372. doi: 10.1007/s11219-015-9292-4

[17] Lazic L, Mastorakis N. Cost effective software test metrics. WSEAS Transactions on Computer 2008; 7(6): 599-619. doi: 0.1007/s10009-014-0332-3

[18] Swinkels R. A comparison of TMM and other test process improvement models. MB-TMM Project Report. Frits Philips Institute, 2000.

[19] Veenendaal V. Guidelines for testing maturity. Professional Tester 2002. 3(2): 21-24.

[20] Neto AD, Subramanyan R, Vieira M, Travassos GH. Characterization of model-based software testing approaches. In: Proceedings of the 1st ACM International Workshop on Empirical Assessment of Software Engineering Languages and Technologies: Held in Conjunction with the 22nd IEEE/ACM International Conference on Automated Software Engineering (ASE); Atlanta, Georgia, USA; 2007. pp. 31-36.

[21] Burnstein I, Suwanassart T, Carlson R. Developing a testing maturity model for software test process evaluation and improvement. In: IEEE International Test Conference; Washington, DC, USA; 1996. pp. 581-589.

[22] Michael F and Ina S. A taxonomy of risk-based testing. The International Journal on Software Tools for Technology Transfer 2014 16(5): 559-568. doi: 10.1007/s10009-014-0332-3

[23] Black R. Pragmatic Software Testing: Becoming an Effective and Efficient Test Professional. New York, NY, USA: John Wiley & Sons, 2007.

[24] Andersin J. TPI-a model for Test Process Improvement. Department of Computer Sciences, University of Helsinki Finland 2004.

[25] Sommerville I. Software Engineering. 8th ed.USA: Addison-Wesley, 2007.

Appendices

A1. Details of similar surveys

Table S: Related surveys

Country	Description
Australia	*Year:* 2003 − 2005 *Respondents:* 65 *Objectives:* To determine the types of tools, metrics, standards, and techniques *Types of Org. by Sector:* 67.7% Commercial, 15.4% Private, 10.8% Government, 6.2% Public *Types of Org. by Industry:* 49.2% Software houses *Testing Techniques:* 42/65 organizations use Structural software testing with Regression and acceptance as common techniques. *Software Metrics:* 38/65 organization use measurable test objectives as metrics *Standards:* 47 organizations use software testing standards *Test Plan:* 56 organization duly follow standards test plan *Independent Testing Teams:* 44 out of 65 organization possess independent testing team *Testing Levels:* 33/65 organizations adopted Integration testing as testing levels *Testing Tools:* 44 respondents were inclined to automated testing activities *Documentation:* Only 54 organizations document test results for future follow-up. *Testing Budget:* 3 organizations out of 65 allocate 40% of total budget *External Testers:* 24/65 firms hire external testers *Training:* 47/65 organizations provide training *Barriers:* Lack of expertise, time constraint, and cost
New Zealand	*Year:* 2005 *Respondents:* 61 *Objectives:* Comparison between current industry's software testing practices and the proposed framework for testing *Types of Org. by Sector:* 75% organizations are Private *Types of Org. by Industry:* Out of these 75%, 51% are technology companies *Testing Techniques:* 43/61 organizations adopt functional testing with Dynamic testing as common testing technique *Software Metrics:* 31/61 organizations use defect count metrics *Standards:* 51% of the organizations do not follow any standards at all *Test Plan:* 23% organizations did not ever make a test plan *Independent Testing Teams:* 24/61 do not have a separate testing role *Testing Levels:* Mostly Unit testing is performed in organizations *Testing Tools:* 56% Organizations do not use any testing tool *Documentation:* 58% organizations document their testing procedures *Testing Budget:* Nil *External Testers:* Only 8% organizations hire external testers *Training:* 23% organization focus on training from External organizations *Barriers:* Lack of expertise, time constraint, and cost
Canada	*Year:* 2010 *Respondents:* 246 *Objectives:* To determine strengths and vulnerabilities in the software development industry *Types of Org. by Sector:* Customized software development company *Types of Org. by Industry:* Packaged software development company *Testing Techniques:* Performance testing is mostly adopted with boundary value analysis as the most common technique *Software Metrics:* 70 organizations use code coverage as metric

Table S: (Continued)

	Standards: Nil
	Test Plan: Nil
	Independent Testing Teams: 44/65 have independent testing teams
	Testing Levels: Unit testing is mostly adopted as the testing level
	Testing Tools: 100 organizations performs testing manually
	Documentation: Nil
	Testing Budget: Nil
	External Testers: Nil
	Training: Half of the respondents informed that they are getting formal training for software testing
	Barriers: Lack of expertise, time constraint, and cost
Turkey	*Year:* 2013
	Respondents: 202
	Objectives: To determine types of software engineering practices
	Types of Org. by Sector: Government
	Types of Org. by Industry: Military and Defense, IT, telecommunication
	Testing Techniques: TLD (Test Last Driven) Development approach 17 participants rarely used the Test Driven Development (TDD)
	Software Metrics: Code coverage metric, defects per line of code
	Standards: ISO/IEC 9126 Software engineering – Product Quality or IEEE 730-1998 standards for software quality assurance are adopted
	Test Plan: Nil
	Independent Testing Teams: No Independent testing teams
	Testing Levels: Developer themselves perform acceptance testing ,integration testing
	Testing Tools: Mostly manual testing is used

A2. The Survey Form

The survey form is available online:

docs.google.com/forms/d/e/1FAIpQLScQiDKy9WI5h-w0u-jE3gV3Pf_-ulckJodO72eJ1Fj_TSA9nQ/viewform