

An Empirical Study to Review and Revise Job Responsibilities of Software Testers

Tanjila Kanij

Swinburne University of Technology
Hawthorn, Victoria, Australia
Email: tkanij@swin.edu.au

Robert Merkel

Monash University
Clayton, Victoria, Australia
Email: robert.merkel@monash.edu

John Grundy

Swinburne University of Technology
Hawthorn, Victoria, Australia
Email: jgrundy@swin.edu.au

Abstract—The broad domain of software testing includes different job responsibilities such as creating test plans, devising and running a variety of tests, documenting results, to liaising between different development teams. The research described in this paper was conducted to collate and revise a list of software testing job responsibilities by applying three different social research methodologies to collect information from different sources. We found that “test” specific responsibilities are divided into several unit tasks including test suite generation, execution of test plans, and so on. We also found that along with test specific responsibilities, software testers must perform a number of other tasks in order to carry out their roles.

I. INTRODUCTION

The concept of software testing refers to the job of finding faults, checking the fitness and appropriateness, and ensuring a certain level of reliability of software systems before their deployment. In this process, the tester searches for instances where the software does not perform according to its specification - that is, software “bugs” - and tries to identify as many as possible within the limits of the resources available to them.

In identifying “bugs”, software testers typically start with a test plan and then design suitable tests according to the plan. They execute the tests and list the bugs found, including where possible location, characteristics and sometimes potential cause. These bugs also need to be reported to the responsible team of developers in order to be fixed. Software testers then re-test the reported bugs after they are reported to be fixed. This generalised cycle of software testing may include many more unit tasks. However, a comprehensive list of such unit tasks is, to our knowledge, unavailable. This limitation was found while designing a research study to find possible connections between personality and performance of software testers [4]. Without a good knowledge of what software testers spend their time doing it was infeasible to find out who is good at doing what!

The research study described here was thus designed to review revise previously developed partial lists of the unit job responsibilities of software testers. Such a list will be useful for policy makers and recruiters to design new job descriptions for software testers. It will also help young graduates to have an idea of software testing profession and will help them in deciding career objectives. It will allow us to further investigate different testing roles and responsibilities and potential impacts of tester personality, organisational culture, team climate, and other factors.

The rest of the paper is organized as follows: Section II describes our research design. Section III presents the results of this study and Section IV discusses possible threats to the validity of the research, Section V discusses key findings and Section VI concludes the paper.

II. METHODOLOGY

This research is designed to address the question “What unit tasks does software testing include?”, focusing on the human tester aspect of software testing, rather than techniques and tools. One source of the required information is the collection of detailed work logs of software testers. In broad terms, a worklog is collection of units of work performed with a tracking of time spent to perform the unit of work. The analysis of worklogs of instructional design professionals [2] is an example of this type of research.

Another source of this information is the descriptions of software testing roles in job advertisements. Recruiters generally prepare a list of responsibilities to be carried out by the newly recruited person and provide the list in the job advertisement.

Another possible source of information about tester unit tasks is bug repositories. Bug repositories are the collection of descriptions of the bugs encountered in a software development project. Bugs are mostly found while testing software. Thus, the information about what the reporter did to produce the bug describes some of the tasks performed by a tester to find a bug.

Obtaining detailed work logs from working professionals is very challenging and thus we report the collection of worklog using a case study [3] methodology from a small number of testers. Extracting tester unit tasks from job advertisements used a survey [3] technique. Mining bug report repositories for unit tester tasks is also a form of analytical survey [3].

A. Case Study of Software Testers’ Work

In this part of the study we collected the worklogs of several software testers through a customized web interface. Over a period of approximately two weeks, testers indicated (in a broad sense) the nature of each individual task they had worked on.

B. Survey of Software Testing Job Advertisements

In this approach, we collected the job descriptions of software testers from popular job web site monster.com [1] over a period of five days using an automated PHP script.

We chose this site as it has a very large number of software development related positions advertised, from a wide variety of locations, company size and type of industry. The collected job descriptions were then analysed manually to identify a set of unique job descriptions and tester unit task types.

C. Survey of Bug Descriptions in Bug Repositories

There are number of open source bug repositories, and as such the number of available bug reports is huge. To draw a boundary of our search space we limited our search to bug reports of testing related products of Eclipse and Firefox repositories. The products were Eclipse Java Development Tool (JDT) and Firefox (Server software) Testopia. We used an automated script written in PHP to search for all bugs that contained the keyword “test” in the summary and/or in comments fields of the bug reports. The “steps to reproduce” section was extracted from those bug reports for analysis.

D. Analysis

The nature of the data collected in this research was a mix of qualitative from the case studies and quantitative from the job description data extraction and bug log analysis. We applied a “Content Analysis” [3] approach and adopted the following steps to analyse the collected data: (i) testing unit tasks were identified and listed from the complete worklogs, job descriptions and bug descriptions; (ii) the frequency of each unit task was counted; and (iii) the listed unit tasks were analysed to group similar tasks into common categories.

III. RESULTS

A. Case Study of Software Testers’ Worklogs

There were a total of 6 cases (work log participants), from Australia, New Zealand, Germany and Bangladesh. Of them, two were female. Three of the cases were from the 26-30 years age group, two were 41-50 and one was 51-60.

Three cases stated that their main job responsibility was to “test modules/programs developed by others”. Two cases stated that their main job responsibility was “managing testers within a project”. One case stated that their main job responsibility was “testing and managing”. Two cases had more than 5 years of experience and two cases had between 3 to 5 years of experience. The other two cases had 1 to 3 years of experience and less than a year of experience respectively.

Summary of Case Study: A total of 6 studied cases submitted their worklogs for different time span ranging from 2 to 37 working days. We stored these work logs in a mySQL database and used a variety of queries to determine unique strings from the submitted worklogs. A total of 63 unique strings was found from the total submitted job work log tasks. After an manual inspection, we eliminated the redundant responsibilities from the list and obtained a list of 44 unique work tasks. These 44 unique tasks were analysed carefully and by grouping similar jobs together we found 12 distinct categories of jobs were performed by the cases. The 12 categories along with the unique tasks in each category are listed in Table I. The last column of the table represents the frequency of unique tasks.

B. Survey of Software Testing Job Advertisements

We found 47 job advertisements for software testers on monster.com over 5 days. We extracted unit tasks from these job advertisements, performed a manual inspection and found 39 unique tasks. By grouping similar jobs together, we found 11 distinct categories of job. The categories with the unique tasks are reported in Table II.

C. Extraction and Analysis of Bug Descriptions

The steps to reproduce a bug as described in bug reports were very specific about testing activities. We identified 24 unique tasks from the bug reports. Most of those were testing related and as such were grouped as- “Testing”.

The jobs in the “Testing” category were subdivided into small groups. Management of test plans was put in the testing category (as different to the other two methods described in Section III-A and III-B). The reason is that unique tasks, such as prepare test plan, cloning test plan and so on, were specific plans for one test execution in this case. The jobs identified from the repositories are reported in Table III.

IV. THREATS TO VALIDITY

There are number of threats that can affect the validity of our findings. We discuss some of those here. Some responsibilities may fall outside the time frame of the period worklog was collected for, may not be mentioned in the collected job advertisements or in the bug descriptions. This is a common threat of this type of research. To overcome the threat we need to collect worklog, job advertisement and bug descriptions for entire project life cycles across many different projects, which was not feasible. However, to minimize the effect of this threat our plan was to collect as much data as was feasible. The information we are seeking to extract from this research is highly dependent on the organization, type of software being developed and the development process being followed. We tried to capture this information in collecting worklogs. However, in the other two methods this information was infeasible to obtain. Thus our finding do not ensure the same tasks are practiced in organizations operating in different business domains and using different development processes.

V. DISCUSSION

A unified list of unit tasks extracted from all sources can be found at: <http://testingsurveys.org/Worklog/UnifiedTaskList.pdf>

From our research, we found that “Test” specific responsibilities of a software tester comprise a number of unit tasks, such as creating test cases, preparing environment, executing test cases and so on. These unit tasks were found from all three methods followed in this research study. As a consequence of testing testers need to report results. We observed that testers also need to explain and analyse the test reports. They also perform other writing tasks such as preparing test documents, writing test status reports. The amount of writing and reporting related tasks performed by the testers indicate that they need good communication and writing skill.

We also found that software testers are engaged in many research and development types of tasks such as requirement analysis. In addition, testers also gain domain knowledge and conduct research about testing tools and new technologies.

TABLE I: Summary of case study

Category of tasks	Unique tasks	Frequency of occurrence
Planning	Resource planning	6
	Planning upcoming test phases	1
	Planning test orders for next financial year	1
Research and Development	Looking for test automation tools that will meet our requirements	1
	Analysing requirements and functional designs	38
	Gaining domain Knowledge	2
	Finalizing proof of concept on automated tools	2
	Consulting with Subject Matter Experts	1
	Discussion of change in direction for another project - proposal to provide system testing support	1
Review	Peer Review work done by another tester	13
	Human Resources Responsibility - annual review of tester on other project	2
	Review defects logged by other testers	2
	Document reviews	1
Testing	Setting up test environment	4
	Develop test specification	7
	Develop test suite	35
	Execute tests	29
Retesting	Retesting of fixed defects (not execution of regular test)	5
	Defect retest	1
Reporting	Reporting test results	7
	Explaining test results	5
	Turn Around Report - status of all tasks currently with test team	5
Debugging	Debugging	22
Main tenance	Maintaining test infrastructure	25
	Upgrade test tools, versions, environments	2
Managerial	Coordinating testing team	15
Adminis trative	Prepare and provide quote for testing (preparation and execution) for a change request	4
	Preparation work for upcoming test phases	1
	Checking accounting for system test efforts	1
	Admin tasks related to Test Lead	1
	Various administrative tasks	2
	Meeting	Test team meetings
Meeting	Project team meetings	17
	Development Practice Meeting	1
	Company meeting	4
	Meeting with external stakeholder	2
	Help and Support	Helping other testers
Help and Support	Help developers with existing functionality	1
	Support for support team	4
	Provide support to trainers	1
	Miscel laneous	Waiting for development to fix defects
Miscel laneous	Work for company I work for (as distinct from company where I am contracted to)	5
	Work for Test Practice Group	1
	On Leave	1
	Total	299

Testers also participate in preparing quality standards and in improving process.

Software testing tasks also include debugging tasks, such as finding the root cause of the defect, exploring and suggesting possible resolutions. This was evident from both the worklog and the job advertisements. To find the cause and the possible resolution of defects testers need to review the code.

Along with the testing and debugging responsibilities testers (mostly senior level such as managers) perform some managerial, supervision and planning related jobs. Testers also must collaborate with other stake holders, such as programmers, managers and clients, and participate in meetings. The types of meetings were not clear from the job advertisements. However from a review of the collected worklogs we see that they participate in different types of meetings such as project team meeting, test team meeting, company meetings and so on. This indicates a significant amount of testers' time is allocated to meetings. Hence a different category called meeting is formed from the analysis of the worklogs.

VI. CONCLUSION

We adopted three different research methods to collect information from different available sources and prepared a

list of unit tasks performed for testing. We found that along with testing many other unit tasks such as debugging, planning, maintenance, managerial and collaboration with others are performed by testers. We believe the identified category of jobs along with specific unit tasks of each category will help recruiters to design job responsibilities for testers. it will also help young graduates get better understanding of the responsibilities of this role in selection of their career choices. In addition, this research may assist in the development of tester training and education.

REFERENCES

- [1] Recruitment website. <http://www.monster.com/>. [Online; accessed 07-July-2010].
- [2] S. Cox and R. T. Osguthorpe. How do instructional design professionals spend their time? *TechTrends*, TechTrends.
- [3] M. Denscombe. *The Good Research Guide for Small-Scale Social Research Projects*. Open University Press, Milton Keynes, UK, 2003.
- [4] T. Kanij, R. Merkel, and J. Grundy. An empirical study of the effects of personality on software testing. In *Accepted to be published at 26th Conference on Software Engineering Education and Training*, 2013.

TABLE II: Summary of job descriptions

Category of tasks	Unique tasks	Frequency of occurrence	
		Eclipse JDT	Firefox Testopia
Testing	Prepare test plans	35	
	Develop test specification	3	
	Write test cases	33	
	Create test input	4	
	Execute test cases	43	
	Implement and manage automated testing package	40	
	Retest after defects are fixed	4	
Research and Development	Develop and maintain quality standards	20	
	Gain knowledge of product	6	
	Requirement analysis	12	
	Suggest enhancements to test mechanisms	2	
	Identify issues during new product development	1	
	Learn new technologies	1	
	Participate in process improvement	11	
	Determine resource needs	1	
Writing and Reporting	Design new test strategy	9	
	Report test results	24	
	Analyse testing report	16	
	Track, record and report testing status	13	
Debugging	Write test documents	10	
	Determination of the cause of defect	3	
Assessment and evaluation	Find and recommend options to resolve testing issues	11	
	Assess code coverage	1	
Planning	Evaluate usability	2	
	Prepare and ensure timeline deliverables	11	
Managerial	Participate in budget process	1	
	Manage test team	4	
Supervision	Setup and maintain test infrastructure	11	
	Review others' work	21	
Administrative	Provide training	8	
	Perform software configuration	6	
	Defect tracking	6	
	Administrative work	7	
	QA Review	2	
	Release management	3	
Collaboration	Track and control change management process	2	
	collaborate with other stack holders	14	
	Participate in status meeting	3	
	Communication with Clients	1	
Total		305	

TABLE III: Summary of bug descriptions

Category	Groups	Tasks	Frequency of occurrence	
			Eclipse JDT	Firefox Testopia
Testing	Preparation	Preparing environment	9	33
		Writing test plan	0	16
		Modify testplan	0	3
	Test plan management	Searching test plan	0	13
		Exporting/cloning test plan	0	7
		Deleting test plan	0	2
		Creating new testcases	3	29
		Modify existing testcase	1	10
	Test case management	Searching/selecting existing testcase	5	18
		Cloning test case	0	7
		Create test run	0	10
		Edit test run	0	1
	Test run management	Selecting test run	0	10
		Cloning test run	0	2
		Linking test plan with test case	0	5
		Linking test case with test run	0	4
	Association of components	Executing test	5	4
		Creating programs for executing tests	1	0
	Test execution	Check output	3	17
		Other	Change test status	0
Write notes	0		1	
Reporting	Create test report		0	7
Research and development	Read help files	0	2	
Planning	Time estimation	0	1	
Total			27	210