UNIVERSITY OF TARTU FACULTY OF SCIENCE AND TECHNOLOGY Institute of Computer Science Software Engineering Curriculum

Yaroslava Malash

Case study in Mobile Testing at Playtech Kiev

Master's Thesis (30 ECTS)

Supervisor: Dietmar Alfred Paul Kurt Pfahl Co-supervisor: Darya Alymova

Tartu 2016

Case study in Mobile Testing at Playtech Kiev

Abstract:

The modern digital market of smartphones is growing every day. The smartphone has become an important part for many people in their everyday social life. People use mobile devices for different activities, including business and personal needs.

Currently, many IT companies started to be focused on mobile development to provide different mobile solutions for the global market. Mobile software testing is a part of the mobile development process. It became an additional part of software testing process. There are a lot of different procedures, models, test cases and approaches within mobile software testing. This thesis described the mobile testing stare-of-practice at QA department, specifically for Mobile and Web Service Team at Playtech Kiev.

As a result, of this case study the "Mobile and Web Service Team" received a report with the structured list of suggestions and possible improvements. The QA manager and QA Team Leaders of the "Mobile and Web Service Team" gave feedback on each provided suggestions, approved the most important points based on the team's priority.

Keywords:

Software testing, case study in software testing, mobile testing research, qualitative study, quantitative study

CERCS: P170

Mobiilse tarkvara testimise juhtumiuuring Playtechi Kiievi

Kokkuvõte

Mobiilsete seadmete turg kasvab igapäevaselt. Nutitelefonidest on saanud tähtis osa paljude inimeste igapäevasest sotsiaalelust. Mobiiltelefone kasutataksse paljudel erinevatel eesmärkidel, isiklike otstarvete kõrval ka ärilistel ja meelelahutuslikel.

Paljud IT organisatsioonid on suunanud oma fookuse mobiiliarendusele, et pakkuda erinevaid mobiilseid lahendusi globaalsele turule. Mobiilse tarkvara testimine on üheks osaks mobiilse tarkvara arenduse protsessist. Sellest on saanud täiendav osa tarkvara testimise protsessist. Mobiilse tarkvara testimiseks on välja töötatud mitemeid protseduure, mudeleid, testimise lahendeid ja lähenemisviise.

Käesolevas magistritöös on kirjeldtatud mobiilse tarkvara testimise protseduuri Playtech Kievi QA osakonna näitel, keskendudes Mobiili- ning Veebiteenuste meeskonnale.

Mobiilse tarkvara arenduse protsess on üles ehitatud kasutades QA juhi poolt esitatud dokumente ning tehnilise nõudeid, töötajate küsitlustikku ning erialast kirjandust.

Antud juhtumiuuringu tulemusena valmis Mobiili- ning Veebiteenuste meeskonna jaoks raport struktureeritud nimistuga soovitustest ning võimalikest parandusettepanekutest. QA juht ning Mobiili- ning Veebiteenuste tiimi QA juht andsid tagasiside igale soovitusettepanekule, kiitsid heaks kõige tähtsamad punktide elluviimise sõltuvalt meeskonna tulevikusuundade ja uurimuste pealt sätitud prioriteetidest.

Märksõnad

Mobiilse tarkvara testimine, juhtumiuuring tarvaratestimise kohta, mobiilse testimise uurimustöö, kvalitatiivne uuring, kvantitatiivne uuring

CERCS: P170

Acknowledgment

I would like to express my deepest appreciation to all those who provided me the possibility to complete this master thesis.

First, I want to thanks to University of Tartu and my Program Manager Marlon Dumas for advices and ability to take extension year to complete my master thesis.

A special gratitude I give my supervisor Dietmar Pfahl for stimulation and suggestions in conducting the case study design and master thesis report.

Furthermore, I would also like to acknowledge with much appreciation Darya Alymova, my co-supervisor for her business time and provided information for the case study research within Playtech Kiev.

A special thanks go to Victor Nessytov, VP of Playtech in Kiev Ukraine for the accepted my proposal about cooperation with Playtech Kiev and the MWS Team during the master thesis. The best mentor I have ever met in my professional life.

Many thanks go my friends Svetlana Omelkova, Sergey Omelkov, John Avila, who always helped and supported me during my study and lifetime in Tartu.

Finally yet importantly, I want to express my gratitude to my special and dear

Roman Leonov, who is always on my side.

Last but not least, a big thanks to my family in Kiev, who always believes in me.

Table of Contents

1.	Introd	luction	8
	1.1	Problem Statement	8
	1.2	Outline	9
	1.3	Research Objectives	9
	1.4	Research Questions	10
	1.5	Context	11
2	Rel	ated work	13
3	Cas	se study design	14
	3.1	Research environment and target group	14
	3.2	Data collection procedure	15
	3.2	.1 Case study protocol	15
	3.2	.2 Meetings with QA manager	16
	3.2	.3 Archival records	16
	3.2	.4 Questionnaire for QA Engineers	17
	3.3	Data analysis	19
	3.3	.1 Quantitative Data Analysis	19
	3.3	.2 Qualitative Data Analysis	19
	3.4	Data validity	19
4	Res	sults	20
	4.1	BPMN model "As_Is"	20
	4.2 Tł	ne results of Quantitative Analysis	20
	4.3 Tł	ne results of Qualitative Analysis	26
	4.4 Fi	ndings from the Data Analysis	29
5	Sug	ggestions	31
	5.1 B	PMN "To_Be" model	31
	5.2 Sı	aggestions on hardware level	32
	5.3 Sı	aggestions on Team Management Level	32
	5.4 Sı	aggestions on Team Level (administrative approach)	33
	5.5 Sı	aggestions on Top-Management Level	34
	5.6 Sı	aggestions from the Literature review and findings	34
6	Fee	back and Impact	35
7	Co	nclusion	41
8	Ref	ferences	42
L	icense		44

Apper	ndix	.45
I.	Glossary	.45
II.	Documents from Playtech Kiev	.46

List of figures

Figure 4.1 BPMN model of "As_Is" process at Playtech Kiev
Figure 4.2.2. How often do you get mobile devices with 0% of battery charge?22
Figure 4.2.3. How much time in average does it take to find a specific mobile device?23
Figure 4.2.4. Do you believe test automation on mobile devices can dramatically decrease time of manual testing?
Figure 4.2.5. What would you change in current mobile testing process?
Figure 5.1 BPMN model of "To_Be" process for Playtech Kiev
List of tables
Table 3.1. Distribution of work experience of participants 14
Table 3.2 Gender distribution of participants 15
Table 3.2.1. Case study protocol 15
Table 3.2.2. QA questionnaire 18
Figure 4.1 BPMN model of "As_Is" process at Playtech Kiev
Table 4.2.1 "Where do you often get a mobile device for mobile testing task ?"
Table 4.2.2 "How often do you get mobile devices with 0% of battery charge?"21
Figure 4.2.2. How often do you get mobile devices with 0% of battery charge?
Table 4.2.2 "How often do you get mobile devices with 0% of battery charge?"
Figure 4.2.3. How much time in average does it take to find a specific mobile device?23
Table 4.2.4 "Do you believe automation on mobile devices can dramatically decrease time of manual testing?"
Figure 4.2.4. Do you believe test automation on mobile devices can dramatically decrease time of manual testing?
Table 4.2.5 "What would you change in current mobile testing process?"

Figure 4.2.5. What would you change in current mobile testing process?	25
Table 4.2.6. "What would you never suggest to do in mobile testing process?"	25
Table 4.3.1. Color coding of complaints from QA Engineers during the software tes process	ting 26
Table 4.3.2. Coding definition of the Qualitative data	28
Figure 5.1 BPMN model of "To_Be" process for Playtech Kiev	31
Table.6.1 Feedback of report	35

1. Introduction

The mobile devices market is growing every day. The smartphone has become an important part of everyday activities for many people. People use mobile devices for different activities, including business and personal needs. As a result, many IT companies started to focused on mobile development to provide various mobile products for the global market.

Mobile software testing is a part of the mobile development process. It became an additional part of software testing process. There are a lot of different procedures, models, test cases and approaches to mobile software testing.

This thesis describes the mobile testing approach used by QA Department within MWS (Mobile and Service Team) at Playtech Kiev

1.1 Problem Statement

Mobile software testing is significantly different from the testing web and desktop applications. It is more complex and different due to:

- Diverse range of mobile devices hardware (screen size, touchpad, hardware configuration);
- Variety of mobile devices (HTC, Samsung, Apple, Nokia, etc.);
- Variety of mobile operating systems (Android, IOS, Symbian, Windows);
- Different version of operation systems (Android 1.0-5.5, IOS 5.x, IOS 6.x, BB 5.x);
- Different mobile network operators (GSM and CDMA);
- Frequent updates (Android and IOS)

Fundamental differences between mobile and desktop testing:

- Mobile device screen size is small;
- Different mobile devices have different screen resolution;
- Mobile devices have less memory than desktop computers;
- Mobiles use network connections;

Test automation on mobile devices is also different and more complex because of the mobile specific hardware and software requirements.

1.2 Outline

The Master Thesis is structured as follows.

• Chapter 1 – Introduction

This chapter gives an explanation about the subject and point of the problem, describes the research objectives that addressed and research questions that become from the problem identified.

• Chapter 2 – Related work

This chapter contains different research papers and similar studies about software testing problems, methods, models that applied in various organizations and proposed by researchers.

• Chapter 3 – Case study design

This chapter gives a detailed structure of the research methods and data analyses and describes the process of collaboration with Playtech Kiev to conduct and report the case study research in mobile software testing.

• Chapter 4 – Results

The chapter presents the process and results of case study research. It outlines the data sources and results of the data analyses.

• Chapter 5 – Suggestions

The chapter presents the list of suggestions based on results.

• Chapter 6 – Feedback and Impact

The chapter presents feedback from QA Department within MWS Team at Playtech Kiev.

• Chapter 7 – Conclusion

The chapter outlines the findings of the case study and proposes a list of suggestion for Playtech Kiev QA department within MWS Unit.

1.3 Research Objectives

The aim of this case study is to define the current mobile software testing process at MWS Team within the QA department at Playtech Kiev and to understand the weakest parts of the mobile testing process by applying the research methodology and case study design approach.

1.4 Research Questions

To achieve the research objectives, set in section "Research objectives", research questions are developed consisting of a main research question and sub-questions.

Main question: How can the mobile testing process be improved for the MWS Teamwithin the QA department at Playtech Kiev?

The outcome of this research is the case study report specifically for MWS Team within QA department at Playtech Kiev. This will be accomplished by answering the following sub-questions:

Sub-question 1: How is the mobile testing process currently applied in the MWS Teamwithin the QA department at Playtech Kiev?

To answer the question, the research will be conducted by weekly Skype-meetings with the QA manager who leads the mobile testing activities at MWS Team within the QA department at Playtech Kiev. The expected outcome of this sub-question is an overview of current state-of-art and visualization "as_is" business process model of mobile software testing.

Sub-question 2: What are the weaknesses of the currently applied mobile testing techniques and tools?

To answer the current sub-question, the research will be conducted by investigation different mobile testing policy and requirements, applied at Playtech Kiev QA department. Based on the archival information, it will help to develop the questionnaire for QA Engineers.

The expected outcome of this sub-question is to understand the main complaints from QA Engineers. These complaints will be considered the weakest part of the mobile testing process.

Sub-question 3:

How can the current practice of mobile testing be changed and improved?

This research question will be answered by using the requirements identified in the previous sub-questions. Also a literature review about the mobile testing process will be conducted and specific suggestions will be proposed.

1.5 Context

Mobile and Web Services (MWS Team) department is focused on providing turnkey integrated online solutions to Casino owners. They are web portals based on the Liferay content management system, which supports integration with multiple Playtech game providers and other services. Using the most outstanding technologies available on the market including WebSocket protocol for efficient full-duplex communication between web server and web client, Content Delivery Network for improving performance and availability of the resources, Single Page Application model for providing seamless end-user experience and much more, the MWS unit creates fast modern web portals.

The department was founded in July 2014 by uniting two units, Portal, and Hub, which had existed till the moment for more than few years each and had both considerable customer databases. Both departments aimed at creating web portals for Playtech customers but one called "Portal" worked with the desktop solutions only while the other named "Hub" provided the mobile-only solutions.

The former product was called Portal 2.0 while the latter was called Hub. The idea was to provide a generic solution called Portal 3.0 with responsive design fitting any customers' needs and including all the features and supported platforms of both Portal and Hub customers.

Currently, MWS Unit is on its way to the defined goals. Portal 3.0 product already supports a lot of platforms including desktop Windows (Chrome, FF, IE9+) & Mac OS (Safari) and mobile iOS (Safari, Chrome), Android (Chrome, Native browser). Customer web sites based on Portal 2.0 and Hub solutions are actively upgraded to Portal 3.0.

90% of mobile testing is performed on real devices. Emulators and simulators are used only in specific cases, mainly when the necessary device model is not yet available on the Ukrainian market. MWS QA Team also works on providing automation solution for functional testing of the MWS portals on all the supported platforms.

Automation coverage rate differs from team to team. And though it measures up to 60% at most the necessity of manual testing is still significant as there are no available tools for automation of GUI^1 and UX^2 .

¹ GUI – Graphical User Interface

² UX – User Experience

The MWS department is one of the biggest units in Playtech Kiev. It works according to the matrix structure. All the specialists are divided into sub-units. The sub-units are cross-functional teams including FE³ & BE⁴ Development, QA, DevOps⁵, Business Analytics representatives each led by a separate Project Manager responsible for software delivery.

And in parallel each direction has a functional or line manager (Development, QA⁶, Delivery, etc.) responsible for professional level and growth of his/her employees.

About 20% of the engineers in the department are working on Core, which is the Portal 3.0 product development. The remaining sub-units, consisting of 5-12 people each, develop customizations for customers. All the customer sites are divided into two types based on Core customization level:

- 1. Projects imply slight customization (CMS^7 configuration + CSS^8).
- 2. Dedicated Projects (imply the development of functionality above Core).

In addition, both Projects and Dedicated Projects teams support customer sites based on MWS⁹ deprecated projects, Portal 2.0 and Hub.

³ FE – Front – End development

⁴ BE – Backend development

 $^{^{\}rm 5}$ DevOps – Development and operation

⁶ QA – Quality Assurance

⁷ CMS – Content management system

⁸ CSS – Cascading Style Sheet

⁹ MWS Team – Mobile Web Services Team

2 Related work

Regarding recent works and study of mobile testing, mobile testing approaches and scientific methods, based on research papers, mobile testing are examined in different aspects. In the work [1] Haller investigates mobile testing approach based on user needs. The paper presents the framework of mobile testing, which uses at Swisscom IT Services, the framework gives different levels of mobile specification and tests, requirements and user need in specific tasks. Another related study [2] offers the test automation approach for mobile testing to support a variety of updates for mobile smartphones. A big part of mobile testing related to native mobile development, i.e. programs develop based on IOS or Android mobile operation systems. The multivariate testing approach presents [3] the mobile testing for the native application, mainly Android and IOS platforms. Other interesting studies [4] proposes the strategy for the mobile testing process on emulators and describe the environment for that approach.

As there are many varieties of mobile applications authors from this research paper [5], try to investigate and generalize the essential validation and verification rules for testing mobile applications. Usability is one of the important parts of each mobile applications, in this paper, [6] authors propose and present different methodology of usability testing for mobile applications.

Another interesting research [7] considers the cross-platform frameworks, which focused on native mobile development, compare their pros and cons and investigations the general rules and amount of testing combinations for native mobile testing.

As many mobile applications are appearing every day, the question of test automation for mobile application considers in the research paper [8] where authors describe the framework for automation tests for the mobile application. Cloud computing is a big research area that affects and changes different approaches and methods in mobile testing.

The survey research [9] presents cloud-testing techniques, which can provide more effective and scalable testing approach. The testing on mobile application is complex and hard-consuming, the authors of this research paper [10] propose the specific test technique for mobile application testing to optimize the number of test cases and time. One more interesting paper [11] considers the testing requirements for mobile applications. Each research papers have similar motivations and statements. However, in current research considers the specific case for MWS Unit at Playtech Kiev, and it's significantly different case.

3 Case study design

The description of the case study design consists of the following parts:

- Research environment and target group
- Data collection procedure
- Data analyses of the collected data from different sources
- Data validity

3.1 Research environment and target group

The case study entitled 'Mobile testing case study based on Playtech Kiev employment" was descriptive, and the research was done in cooperation with Playtech Kiev Quality Assurance (QA) department, specifically for Mobile and Web Services Team (MWS).

The primary goal of MWS Team is to deliver a high-quality mobile product for the gambling industry. In order, to understand the "bottlenecks" of the current mobile testing process at Playtech Kiev the target group of 28 Quality Assurance (QA) engineers participated in current research. The distributions of work experience and gender are shown in Tables 3.1 and 3.2, respectively.

Work Experience at Playtech Kiev (years)	Amount (people)
2-3 years	6
1-2 years	6
More than 3 years	1
Less than 1 year	15
Total	28

Table 3.1. Distribution of work experience of participants

Table 3.2 Gender distribution of participants

Gender of QA Engineers	Amount (people)
Male	20
Female	8
Total	28

3.2 Data collection procedure

The data collection procedure consists of the following parts:

- Case study protocol
- Weekly meetings with QA manager
- Archival records
- Questionnaire for QA Engineers

3.2.1 Case study protocol

Case study protocol presents the data collection plan, communication channels, number of people per data source, meeting durations. The information structured and presented in the Table 3.2.1

Data sources	Participants	Com- ments	Amount of people	Duration
Documentations and requirements policy	QA manager	Skype- meetings	1 – QA man- ager	30 mins per meetings7 meetings during 3 months
QA Leaders Skype - meeting	QA Lead's group skype-chat	QA Lead's group	3 - QA Lead- ers	1 hour Only once

Table 3.2.1. Case study protocol

		video skype-chat		
QA's questionnaire	QA specialists and QA Leaders	The opin- ion from employers	25 QA engineers3 QA Team Leaders	10 mins for each person

3.2.2 Meetings with QA manager

Initially, the collaboration started via Skype with Quality Assurance Manager (QA Manager) at Playtech Kiev MWS department.

During the research, the QA manager described the general workflow of mobile testing, provided information about MWS Team Unit, shared some documents regarding mobile testing requirements, which helped to understand the current software testing process, which applied and used in the QA department within MWS Team at the Playtech Kiev.

The documents are of the following:

- MWS introduction about custom projects and core platform
- Software testing workflow at MWS Team
- Updated documentation about Privacy and Policy at Playtech Kiev
- Access into Confluence for investigation archival records

3.2.3 Archival records

QA Manager provided access into Confluence¹⁰, for investigation archival records, regarding the software-testing process, from the beginning (2013-year) until nowadays, which currently employed and used at Playtech Kiev QA department within MWS Team.

The documents consist of the following:

- Core mobile device support the list of available mobile devices, operation systems (OS) and Browser Support
- "Golden Device" the list of mobile device support
- "Silver Device" the list of mobile device support

¹⁰ Confluence - software, used by teams in collaborative development in IT industry.

- MWS mobile devices document the list of all physical mobile devices at MWS Team
- The requirements workflow in a period from 2013 2015

Golden mobile devices support – means that for this set of devices a thorough User Interface (UI) testing is performed. Any found critical high priority defects will get fixed. Silver mobile devices support – means that for this set of devices only smoke testing is performed. Any found critical high priority found defects will get fixed.

3.2.4 Questionnaire for QA Engineers

The QA engineers interviewed were selected from the employees working at Playtech Kiev QA department. The questions for the Questionnaire created based on the interviews with the QA manager and Skype-meeting interviews with QA Team Leaders. The interview completed by using Google Form spreadsheet and contained five questions about software mobile testing process, including two open questions about complaints and possible suggestions of the improvements for mobile testing process.

The QA engineers filled out questionnaires distributed to them while they were working in various mobile testing activities during their work at Playtech Kiev. The mobile testing activities were engaging with different custom gambling projects and core platform development by MWS Team.

The aim of the QA questionnaire is to identify weakest points of the software mobile testing in the current state-of-art process within QA department, MWS Unit.

The questionnaire is shown in Table 3.2.2.

1. How long do you work at Playtech?					
less 1 year	1-2 years	2-3 years	more than 3 years		
2. How oft	en do you get mobile d	evices with 0% of batto	ery charge?		
Almost all the time	Very often	Often	Not very often		
3. How much ti	me in average does it t	ake to find a specific m	nobile device?		
0-10 min	10-20 min	more than 20 min			
	4. What do you disli	ke more in current proc	ess?		
Lack of mobile de- Time to find a suita- Lack of automation Uncharged m			Uncharged mobile		
vices	ble device	mobile testing	device		
5. Do you believe automation on mobile devices can dramatically decrease time of manual testing?					
Yes No Partly					
6. What would you change in current mobile testing process? (open question)					
7. What would you never suggest to do in mobile testing process? (open question)					

Table 3.2.2. QA questionnaire

3.3 Data analysis

The data analysis consists of the following parts:

- Quantitative Data Analysis
- Qualitative Data analysis

3.3.1 Quantitative Data Analysis

Quantitative Data Analysis deals with numbers measured on interval and ratio scales.

The Questionnaire for QA Engineers created based on meetings with QA manager, Skypeinterview with QA Team Leaders and documentation from Confluence¹¹. The data is gathered and compiled on the Google Excel spreadsheet; The data analysis will be done by using statistical calculations and descriptive statistics. The descriptive statistics will represents what the data show, in this case, what data shows from the QA Questionnaire.

3.3.2 Qualitative Data Analysis

Qualitative data deals with description. For qualitative data analyses will be used a tool - Coding Analysis Toolkit (CAT)¹²

The data analysis will be done using color coding. Each "color code" will have the description, the description helps to identify the answers to the research questions.

3.4 Data validity

The data sources of the current research were provided by the target group of people, who are working with this data and use it in their job activities at Playtech Kiev MWS Team.

The validity of data cannot be checked for 100% due to the geographical location between Kiev and Tartu.

However, these are no reasons to not trust to gathered the information since the case study was conducted for particular QA Department within MWS Unit at Playtech Kiev

¹¹ Confluence – software used by teams in collaboration development in IT industry.

¹² <u>http://cat.texifter.com/</u> - Coding Analysis Toolkit (CAT) software for Qualitative Data Analysis

4 **Results**

The results consist of the following parts:

- BPMN "As_Is" model presents the current state-of-practice
- The results of Quantitative Analysis shows facts and figures
- The results of Qualitative Analysis describes the main points of complaints

4.1 BPMN model "As_Is"



Figure 4.1 BPMN model of "As_Is" process at Playtech Kiev

The visualization of current software testing process, applied at Playtech Kiev QA department depicted in Figure 4. The "As_Is" model developed after the Skype meetings with QA manager.

4.2 The results of Quantitative Analysis

The amount of answers from the QA questionnaire (Table 3.2.2) were collected, tabulated and statistically analyzed. The results presented in relative frequency charts and pie charts for each question.

	Frequency	Rel. Freq.
from colleague	5	18%
from mobile storage	23	82%
TOTAL:	28	100%

Table 4.2.1 "Where do you often get a mobile device for mobile testing task ?"





Table 4.2.2 "How often do you get mobile devices with 0% of battery charge?"

	Frequency	Rel. Freq.
Very often	8	28.5%
Often	11	39%
Not very often	5	18%
Rarely	2	7.25%
Almost all the time	2	7.25%
TOTAL:	28	100%



Figure 4.2.2. How often do you get mobile devices with 0% of battery charge?

	Frequency	Rel. Freq.
0-10 mins	11	39%
10-20 mins	11	39%
more than 20	6	22%
TOTAL:	28	100%

Table 4.2.2 "How often do you get mobile devices with 0% of battery charge?"



Figure 4.2.3. How much time in average does it take to find a specific mobile device?

Table 4.2.4 "Do you believe automation on mobile devices can	n dramatically	decrease time
of manual testing?"		

	Frequency	Rel. Freq.
Yes	17	60%
No	6	22%
Party	5	18%
TOTAL:	28	100%



Figure 4.2.4. Do you believe test automation on mobile devices can dramatically decrease time of manual testing?

	Frequency	Rel. Freq.
Lack of mobile devices	15	54%
Time to find a suitable de- vice	5	18%
Discharge device	4	14.5%
No enough automation test- ing	2	7.25%
Lack of charging units	2	7.25%
TOTAL:	28	100%

T 11 40 C	((1)1) 11	1 .	/ 1·1	
I able 4 / S	"What would	VOU Change in	current mobile	testing nrocess/"
1 4010 1.2.0	millat mould	you onunge m		resting process.



Figure 4.2.5. What would you change in current mobile testing process?

	Frequency	Rel. Freq.
Extend the number of mo- bile devices	12	43%
Always chart the device	5	18%
Do some automation for re- gression tests	3	11%
Do not take work device to home	2	7.25%
Keep the devices in one room	3	11%
do not use password on mo- bile device	2	7.25%

Table 4.2.6. "What would you never suggest to do in mobile testing process?"

Clean the cash after testing	1	3.5%
TOTAL:	28	100%

4.3 The results of Qualitative Analysis

Coding is simply organizing data. Coding also helps to begin the process of systematically analyzing data. Table 4.3.1 shows the results related to the open question from the QA questionnaire. The meaning of color code presented and described in the Table 4.3.2.

Table 4.3.1. Color coding of complaints from QA Engineers during the software testing process

#PERSON	Complaints
1	Support OS that is lower than "current -1"
2	N/A
3	use passwords on mobile devices
4	Provide chargers both for Android and IOS devices for all QA's
5	I do not advise to emulate devices
6	Full regression on all supported devices. It takes large amount of time re- sources though only two main OSes exist. Testing on all devices needed for main functionality. Not full.
7	Testing on old devices and testing on old platforms
8	N/A
9	 change Apple ID upgrade device without manager approve use it in personal needs
10	Automation testing is not useful current projects.

11	N/A		
12	do not take corporate device to home		
	• do not close device in personal drawer and hide a key		
13	• Make sure devices are available for everyone in case of your		
	absence;		
	• Make sure devices are stored properly when shift is over;		
	 Make sure devices are used carefully. to start tasting process on devices with too short period 		
14	 to start testing process on devices with too short period to keep a large amount of devices in one room without returning 		
	it to device storage		
	 to have passwords on devices 		
	• to use personal accounts on devices		
15	• Don't use devices in personal purposes (play games, take photos		
15	etc).		
	Retain files after testing		
16	• do not set passwords on mobiles . it's a waste of time - entering		
	it, keeping it in memory, asking someone what's the pass.		
	• downloading apps (for personal use), that doesn't relate to testing		
	(maps, instagram, etc). I guess, it's better to interface to be clean,		
	so you won't spent time on midnig app you need		
17	to cover plenty devices for exceptionally functional test cases		
18	N/A		
19	not to install updates while testing		
20	check device type on IMS		
21	Use only automation for regression.		
	· ·		
22	N/A		
23	testing on devices with small screen		
	to star of devices with shart bereen		
24	To leave device uncharged and unmarked on confluence		
	To leave device uncharged and unmarked on confidence		
25	do not use password on mobile device		
23	do not use password on mobile device		
24			
20	charge the mobile device after testing		

27	N/A
28	N/A

Initially, all complaints taken from the QA questionnaire without corrections, so the descriptions and answers contain misspellings. Also, some people did not left complaints, as they work at Playtech Kiev less than a year.

The full description and usage of the color code described in the Table 4.3.2. The colors grouped by the amount of mentions from the QA questionnaire.

ID	Color code	Brief definition	Men- tions	Full descrip- tion	When to use
1	Personal Needs (red color)	Using mobile de- vices not for work tasks.	5	Using devices in case of so- cial app, per- sonal app, files, taking home, etc.	In case of personal usage of the mobile device, personal in- stallation different application and files after testing
2	device un- charged (purple color)	Do not charge de- vice after testing	4	Using un- charged de- vices for mo- bile testing.	Spending time for charging mobile de- vices, do not have a place for charging all units of devices
3	lack of de- vices (green color)	Lack of device	4	The Amount of devices which needed to be tested	When lack of mobile devices stopped the mobile testing activi- ties.

Table 4.3.2. Coding definition of the Qualitative data

4	Automa- tion testing (yellow color)	automation test activities	2	Using automa- tion testing for mobile testing is not usable.	When automation tests do not help to improve coverage of test cases on mobile devices.
5	passwords on mobiles (pink color)	mobile device is locked	5	Using pass- words on any mobile de- vices.	When mobile unit has a password and needs to be unlocked for next testing activities
6	emulate devices (orange color)	emulator and simulator tools	1	Using emulator tools and pro- grams for mo- bile testing process	When mobile testing needs to be tested and using only on simula- tors or emulators in case of mobile testing activities

4.4 Findings from the Data Analysis

This section discusses and summarizes the results from the QA questionnaire (Table.3.2.2); the section is divided into categories by the questions from the QA questionnaire.

Frequency of using mobile devices in mobile testing

The first scope of the results is how frequently QA Engineers find the mobile device for the mobile testing and how often the mobile device is ready for the mobile testing, i.e. fully charged.

The results to these questions found that most QA Engineers:

- 82.1% of QA Engineers get the mobile device from mobile storage
- 39% + 39% of QA Engineers spend 10-20 minutes, to find a suitable mobile device for mobile testing activities
- 39% of QA Engineers often get mobile device without charging
- 54% of QA Engineers complaints about lack of mobile devices

Area needs more emphasis

Most QA Engineers (60%) answered that mobile automation testing can dramatically decrease the time of manual testing.

However, the results show that 21.4% of QA Engineers do not agree and partly agree only 17.9%.

The results cannot be considered as the final due to additional time resources for setting up automation mobile testing tools and support for them.

This area needs more deep investigation and research of automation process.

Frequency of advice given

Because of this area, QA Engineers provided a suggestion to extend the number of mobile devices (48%), and 18% of QA Engineers suggest to keep the device charged.

The results of the study, from section 4.4 presents the two main problems of the software testing process within QA Department at Playtech Kiev MWS Team.

- Lack of mobile devices
- Uncharged mobile devices

Based on the complaints listed above, the list of suggestions was provided in order to improve the mobile software testing for QA Department at Playtech Kiev.

5 Suggestions

The sections presents the list of suggestions on different levels, listed below, specifically for QA Department within MWS Team at Playtech Kiev. Each part of suggestions based on the QA Engineers complaints, suggestions and literature findings.

The suggestions grouped as follows:

- BPMN process model "To_Be"
- Suggestions on Hardware level
- Suggestions on Software level
- Suggestions on Team Leader level (administrative approach)
- Suggestuins on Team Managment
- Suggestion on Top-Level management
- Suggestions from the Literature findings

5.1 BPMN "To_Be" model



Figure 5.1 BPMN model of "To_Be" process for Playtech Kiev.

Figure 5.1 shows the proposed "To_Be" process model of the mobile software testing process for Playtech Kiev QA department and MWS Team.

The model provides suggestions for collecting and saving all mobile devices and charging units in one place – "Test Labs". It avoids the problem of charging and searching activities for QA Engineers.

5.2 Suggestions on hardware level

"Test Lab" – take all device in one place, so everybody will know, where to find it.

- Mobile devices (provided by Playtech Kiev) should be stored in one place / room / table / shelf
- Charging unit for each devices on one shelf/table/room
- Each owner should have his/her charging unit
- All devices should be structured based on Android version and IOS version on different shelves and they are not mixed

Personal devices should be in one place at room (charged as well on the separate shelf)

Benefits:

- In similarity with Server room, Test Lab is suitable for managing mobile activities in one place
- Good start for future extending
- Resource allocation in one place

5.3 Suggestions on Team Management Level

Each QA Team leader should include the amount of devices into development planning.

- Control mobile test tasks selection and preparation of devices based on priority and planning
- Use label such as "golden list", "silver list" in order to do filtering in Jira. via <u>JQL</u> (Jira Query Language). It helps to check amount of mobile testing, severity and main priority what should be tested in the first place
- For regression, testing and smoke testing "labels" are also useful in order to control and manage tasks and bugs.
- The "label" policy should be specified and confirmed with all QA Leader, the exception is only allow for Automation QA engineers

Benefits:

- QA management process optimization and partly automation in order to check the amount of work and control them using JQL
- reporting best practice
- Tasks delegation and resource planning
- Control on amount of tasks bugs per projects

5.4 Suggestions on Team Level (administrative approach)

The list of suggestions presents possible approaches for each QA Team Leaders, which can improve the current state-of-art of mobile testing at Playteck Kiev QA Department.

- QA engineers should do the mobile testing activities at Test Lab when it is suitable for mobile testing activity and device specification.
- The Policy of device updating and upgrading should be strict and managed by Team Leader / QA Manager at least once a month.
- Sprint development planning should include a number of mobile devices for future testing activities
- Confluence page (or better electronic table) should be updated based on full amount of mobile devices. It helps to understand which device should be:
 - upgraded
 - removed
 - o fixed
 - extended with new devices
- Confluence page (or better electronic table) can be divided into (tabs) by vendor categories or OS categories
- The owner of this Confluence page (or better electronic table) should be the only one person (preferably QA manager or QA mobile team lead)
- The basic policy (strict as bug report policy) for using devices:
 - testing device at "Test Lab"
 - o delete not useful information from mobile devices after each testing
 - always charge the mobile device

Benefits

- Rules for checking and control of mobile testing
- Saving time for preparation the mobile device for work
- Updating information about all devices (protocol)
- Testing process optimization (garbage in ,garbage out)

5.5 Suggestions on Top-Management Level

- Consider possible cons and pros about Playtech Kiev Test Lab unit within MWS Team
- Consider risks and possible values of development specific for Test Lab within Playtech Kiev

Benefits:

- IT best practice for mobile development and mobile testing
- Manage hardware resources (mobile, devices, tablets, charging units, etc)
- Audit of mobile activities

5.6 Suggestions from the Literature review and findings

One of the useful case for mobile testing approach is to do testing, on mobile devices for Android Platforms, on emulators for IOS platforms, in many cases, emulators of IOS platforms are similar with real devices. The mix of different testing on emulators and simulators can partial solve the problem of lacking mobile devices.

Some companies also suggest using cloud platforms for mobile testing; however, the mobile testing cloud platforms are expensive and not always can cover the completely mobile testing functions.

6 Feedback and Impact

Playtech Kiev QA manager - Darya Alymova got a report with suggestions and possible improvements for the future workflow of mobile software testing process for QA Department within MWS Team at Playtech Kiev.

The suggestions and proposed improvements considered and reviewed by QA Manager. The QA manager validated there suggestions based on Playtech Kiev business and development flow and policy. The results of the feedback show in the Table 6.1

Suggestion on hardware level				
Nuance	Suggestions	Reviewers' comments		
Positive	Each owner should have his/her charging unit	After all the team complained that there is a problem with chargers, we ordered a set of chargers for each of QA specialists.		
Neutral	Mobile devices (provided by Playtech Kiev) should be stored in one place / room / table / shelf	Initially, we had the approach with a single device storage place. It worked for us for some time, but as the number of people in the department is big, we have been ordering each new device model which should be supported by our product in 3 copies. This is needed to ensure there is at least one device for 10-12 QAs.		
		Recently we have divided our devices between 3 independent storages which are placed in different parts of the office for geographical convenience		
Positive	Charging unit for each de- vices on one shelf/ta- ble/room	 The idea is good but its implementation is on hold now for the following reasons: 1. We have only 1 device container which has enough place and correct construction for each of the devices to be placed in it with a standard charger. This container is aimed for automation team purposes. 2. The mentioned above 3 device storage places are simple office drawer units with 		

Table.6.1 Feedback of report

		no option to have chargers for each of devices inside. We have reviewed the option of wireless chargers but we refused them asa. They are expensiveb. They are good if they stay static. In case we have withdrawable drawer they won't stay on their place and won't be efficient.
Positive	The stickers may be useful in order to create so-called "card-index" for better vis- ibility	Stickers with Device Model and ID are Playtech-wide standard. We don't add stickers with OS version as this information is dynamic and it is stored on our tracking Confluence page. Usually when a person searches for a device s/he knows its characteristics.
Neutral	Personal devices should be in one place at room (charged as well on the separate shelf)	This is not a priority for us to handle peo- ple's personal belongings. However, we ask to run testing only on company devices.
Positive	All devices should be structured based on An- droid version and IOS ver- sion on different shelves and they are not mixed	We separated iOS devices from Android devices. However, as the number of withdrawable shelves is 3 in our drawer units it's not possible to have a separate shelf for each version, even major version not taking into account minor ones.
	Administrative approach (Suggestions on Team Level)
Negative	QA engineers should do the mobile testing activi- ties at "Test Lab" when it's suitable for mobile testing activity and device specifi- cation.	QA Engineers execute testing on their work places.a. There is no possibility to have a dedicated place called "Test Lab".b. It is much better to run testing in the environment where QA engineers are close to the rest of their teams and can raise questions within testing activity.
Positive	The policy of device up- dating and upgrading	We have had a strict policy: If a team needs a device to be upgraded its QA Lead must get a written confirmation

	should be strict and man- aged by Team Leader / QA Manager at least once a month.	from the rest of QA Leads. Then he should upgrade the device and track changes on Confluence. Each QA Lead who is asked to provide an approval in his/her turn should ensure the current OS version is not critical for any of his/her teams before giving the written con- firmation to his/her colleagues.
Positive	Sprint planning should in- clude a number of mobile devices for future testing activities	Not all the teams use iterative approach. So, we don't speak about sprint planning. We say that each Release includes func- tional scope & device support requirements. Based on the above and time limitations QA Team selects some specific devices for test- ing and describes this in Test Plan report.
Positive	Confluence page (or better electronic table) should be updated based on full amount of devices. It helps to understand which de- vice should be: upgraded removed fixed extended with new devices	This has been a common practice for us to keep our Confluence page up-to-date. This was one of the first traditions to appear as we bear financial responsibility for devices. We don't select another electronic table for- mat as Confluence allows us to manage an effectively. Access. Only current employees have access to this page. We don't need to add / update / delete email addresses or an- ything like that. b. History management is quite convenient for us. c. Update conflicts are handled correctly. If several people try to update the page at the same time, Confluence warns them and en- sures the changes are not missed.
Neutral	Confluence page (or better electronic table) can be di- vided (tabs) by vendor cat- egories or OS categories	Our table has the following fields Device name Model OS OS Version

		Playtech ID (unique identifier inside the
	Company) Default storage (as we have 3 storages now)	
		Current Owner
		You can sort the table by each of the fields.
Negative The owner of this Conflu- ence page (or better elec- tronic table) should be the only one person (prefera- bly QA manager or QA mobile team lead)	The owner of this Conflu- ence page (or better elec-	It is very hard and not efficient if only I per- son can update the page.
	tronic table) should be the only one person (prefera- bly QA manager or QA mobile team lead)	We have strict rules on updating the page and bearing responsibility for absence of up- dates.
	moone team tead)	Also Confluence manages update conflicts well, as I mentioned above.
		Our rules:
		The current owner specified in the table is responsible for the device from financial perspective.
		When a device is kept in the storage it has the default "Device storage" status. In this case Device Storage Key Keepers are re- sponsible for the device from financial per- spective.
	If no owner is specified in the table for any reason, the person who cleared up the value is responsible.	
Negative	The basic policy (strick as bug report policy)for using devices:	
	testing device at "Test Lab"	
	delete not useful infor- mation from mobile de- vices after each testing	
	always charge the mobile device	
Suggestion on Team Leader level		

Positive	Control mobile test tasks selection and preparation of devices based on prior- ity and planning	QA Leads control test scope and test devices within test planning activity, as mentioned above.
Neutral	Use label such as "golden list", "silver list" in order to do filtering in Jira. via JQL (Jira Query Lan- guage) It help to check amount of mobile testing, severity and main priority what should be tested in the first place	 The idea is good but its implementation is questionable for the following reasons 1. Golden and Silver lists are dynamic. And defects may be put to backlog for a long time. Which means that the labels should be reviewed periodically. 2. Defects which are reported are sometimes reviewed on 1 device, sometimes on several devices. This can be specified in Defect description or a separate Jira custom field (depends on a team). It is more efficient to have information about the specific devices, not lists they refer to in the specific moment of testing.
Neutral	For regression, testing and smoke testing "labels" are also useful in order to con- trol and manage tasks and bugs.	We specify the device models, OS, OS ver- sion as a part of defect description for the reasons listed above.
Neutral	The "label" policy should be specified and confirmed with all QA Leader, the ex- ception is only allow for Automation QA engineers.	We specify the device models, OS, OS ver- sion as a part of defect description for the reasons listed above.
Suggestion: Top-management level		
Negative	Consider possible cons and pros about Playtech "Test Lab" unit within MWS Team	Pros: Devices are easier to manage in 1 place Cons: There should be a fully dedicated person to manage a Mobile Lab There should be an infrastructure organized

		The effectiveness of testing which is run on the regular work place is higher as people communicate with the rest of team members Taking into account the factors above, I think the idea is not applicable for MWS
Suggestions from the Literature review and findings		
Positive	Suggestions from the Lit- erature review and find- ings	We use this approach already. Except test- ing on real devices we use a. iOS simulators. Proved to have a low ratio of false positive and false negative results, which is why we sometimes use them in cases, we have no real device - OS version combinations. Con: Do not support Chrome. b. Android emulators in Chrome Web Dev Toolbar. Have a high ratio of false positive and false negative results because the only thing that is changed is user agent. We run only specific types of tests using emulators. E.g. we use them to get logs from Console. Also we check hex colors if any visual ele- ment is changed.

Summary from the QA manager of QA Department within MWS Team at Playtech Kiev

"MWS QA Team is grateful for the research which has been held. It demonstrated areas for infrastructure and process improvements which we managed to enforce.

E.g. Thanks to the QA Questionnaire, we calculated approximate idle time spent on device search and charging and got approval to order more devices and have 1 chargers set per QA. I believe, a lot of important and interesting ideas are suggested as a result of the research. Not all of them are applicable in the specific environment of MWS department for different reasons. Some of them have already been implemented in a different way. But the value of the suggestions is still big"

Darya Alymova, QA Manager at Playtech Kiev MWS Team

7 Conclusion

The current case study was conducted specifically for the QA Department within MWS Team at Playtech Kiev.

As a result, of this case study the "Mobile and Web Service Team" received a report with the structured list of suggestions. The QA manager and QA Team Leaders of the "Mobile and Web Service Team" gave feedback on each provided suggestions, validated each approach and approved the most important points based on the team's priority.

As a conclusion, the feedback of each suggestion on suggestions levels, listed in the Table.6.1 got positive, negative and neutral points.

Positive point (9 points from feedback) – means that the provided suggestions is valid to use or already used

Neutral point (6 points from feedback) – means that the provided suggestions not in a priority and do not consider important for now.

Negative point (4 points from feedback) – means that the suggestions are not applicable to implement or use

The current study might be use as a background platform for future improvements of mobile testing process at IT companies.

8 References

- K. Haller, "Mobile testing," ACM SIGSOFT Software Engineering Notes, vol. 38, no. 6, pp. 1-8, 2013.
- [2] Gao J. et al, "Mobile application testing: a tutorial," *Computer*, vol. 4, pp. 46-55, 2014.
- [3] H. P. Holzmann C., "Multivariate Testing of Native Mobile Applications," *Proceedings of the 12th International Conference on Advances in Mobile Computing and Multimedia*], ACM, 2014, pp. 85-94.
- [4] Delamaro M. E., Vincenzi A. M. R., Maldonado J. C, "A strategy to perform coverage testing of mobile applications," *Proceedings of the 2006 international workshop on Automation of software test*, ACM, 2006, pp. 118-124.
- [5] Muccini H., Francesco A. D., Esposito P, "Software testing of mobile applications: Challenges and future research directions," *Automation of Software Test (AST), 2012 7th International Workshop on*], IEEE, 2012, pp. 29-35.
- [6] Zhang D., Adipat B, "Challenges, methodologies, and issues in the usability testing of mobile applications," *International Journal of Human-Computer Interaction*, kd. 18, nr 3, pp. 293-308, 2005.
- [7] T. Paananen, "Smartphone Cross-Platform Frameworks: A case study," Jyväskylä ammattikorkeakoulu, 2011.
- [8] Nagowah L., Sowamber G, "A novel approach of automation testing on mobile devices," *Computer & Information Science (ICCIS)*, 2012 International Conference on, IEEE, 2012, pp. 924-930.
- [9] Baride S., Dutta K , "A cloud based software testing paradigm for mobile applications," ACM SIGSOFT Software Engineering Notes, kd. 36, nr 3, pp. 1-4, 2011.
- [10] Liu Z., Gao X., Long X, "Adaptive random testing of mobile application," *Computer Engineering and Technology (ICCET)*, 2010 2nd International Conference on, kd. 2, pp. V2-297, 2010.

[11] Dantas V. L. L. et a, "Testing requirements for mobile applications,"*Computer and Information Sciences, 2009. ISCIS 2009. 24th International Symposium on*, IEEE, 2009, pp. 555--560.

License

Non-exclusive licence to reproduce thesis

I, Yaroslava Malash (date of birth: 12.03.1985),

I, here with grant the University of Tartu a free permit (non-exclusive licence) to

reproduce, for the purpose of preservation, including for addition to the DSpace digital archives until expiry of the term of validity of the copyright

"Case Study in Mobile Testing at Playtech Kiev", supervised by Dietmar Alfred Paul Kurt Pfahl, Darya Alymova

2. I am aware of the fact that the author retains these rights.

3. This is to certify that granting the non-exclusive license does not infringe the intellectual property rights or rights arising from the Personal Data Protection Act.

Tartu, 2016

Appendix

I. Glossary

QA

Quality assurance (QA) the process used to define and assure the quality of a product

QA Engineer

The person whose job is related to quality assurance process

Functional testing -

is a quality assurance (QA) process of testing functional components of the

Application.

Regression testing

is a quality assurance (QA) process that double check that software that was previously developed and tested still performs correctly

CMS (Content management systems)

is a set of processes and technologies that support the collection and managing of information in any form in the Internet

HTML

Hyper Text Markup Language - system for annotating a document in the web browser

CSS

is a style sheet language used for describing the presentation of a document in the web browser

II. Documents from Playtech Kiev

Document name	Document description
Core mobile device sup- port.pdf	The list of available mobile devices, operation systems (OS) and Browser Support specifically for Core Platform mobile testing activities.
Core device support for new features and regres- sion testing.pdf	The list of available mobile devices for implementation new features on code platform for gambling products.
MWS Devices.pdf	The full list of mobile devices at MWS Team
Automation Tests.docx	The list of Core platform components which covered by au- tomation tests
Device Matrix.xlsx	The spreadsheet of all devices and their technical character- istics, which supports in combination for specific customer needs.
Mobile test evolu- tions.docx	The archival records of the mobile process status and updat- ing requirements during 2013 - 2016
Feedback from QA Man- ager.xlsx	The feedback about provided report of suggestions