TALLINNA TEHNIKAÜLIKOOL
Infotehnoloogia teaduskond
Tarkvarateaduse instituut

Maria Kuznetsova 154830

GAME FUNCTIONAL TESTING ON A DERIVCO ESTONIA LTD COMPANY EXAMPLE

Bakalaurusetöö

Juhendaja: Inna Švartsman
MSc

Tallinn 2018

Author’s declaration of originality

I hereby certify that I am the sole author of this thesis. All the used materials, references to the literature and the work of others have been referred to. This thesis has not been presented for examination anywhere else.

Autor: Maria Kuznetsova

21.05.2018

Abstract

This bachelor's work examines the functional testing of the game on the Derivco Estonia Ltd example. The author performs functional testing manually. The aspects of the game must be covered by tests that help to check maximally different parts of the product and their functionality in the testing cycle.

The purpose of this work is to test the specific functionality of the online slot casino game in the final testing phase and after results analysis to conclude how qualitative product is at this stage, what are the most problematic functional areas to put more emphasis in the next testing phase.

This work examines the testing processes, the principles of manual testing, different testing methods and types.

As a result of the work, is possible to see how qualitative the product is at this phase of testing and what are the most problematic parts that need more attention in the future testing. Based on the results analysis, the author makes recommendations for the improvement of testing process and how to make it more effective.

The thesis is written in english and contains 44 pages of text, 7 chapters, 5 figures and 8 tables.

Annotatsioon

Mängu funktsionaalne testimine Derivco Estonia OÜ näitel.

Käesolev bakalaureuse töö uurib mängu funktsionaalset testimist Derivco Estonia OÜ näitel. Autor teostab funktsionaalset testimist manuaalselt. Mängu aspektid peavad olema kaetud selliste testidena, mis aitaksid kontrollida erinevad tarkvara osad ja nende funktsionaalsust maksimaalselt testimise tsükklis.

Töö eesmärgiks on testima online slot kasiino mängu teatud funktsionaalsust finaalse testimise faasis ja edaspidi analüüsides saadud tulemusi järeldada kui kvaliteetne toode on selles etappis, millised on problemaatilisemad funktsionaalsed osad ning kuhu oleks vaja rõhutada rohkem tähelepanu järgmises testimise faasis.

Töös uuritakse testimise protsesse, manuaalse testimise põhimõtteid, erinevad testimise metoodikaid ning tüüpe.

Töö tulemusena näeme, kui kvaliteetne on mäng käesolevas testimise etappis ning problemaatilised osad, millele peab pöörata rohkem tähelepanu edaspidises testimises. Tulemuste analüüsi põhjal Autor teeb soovitusi testimise protsessi paranemiseks ehk kuidas seda effektiivsemaks muuta.

Lõputöö on kirjutatud inglise keeles ning sisaldab teksti 44 leheküljel, 7 peatükki, 5 joonist, 8 tabelit.

Dictionary of abbreviations and definitions

Achievements	Achievements plaque appear when a successful symbol win occurs and the associated symbol pay on the Paytable must highlight.
Ad-hoc testing	A type of software testing, which is performed without planning and documentation, testing is performed by improvisation.
Autoplay	Game feature what enables you to play a set number of spins without having to click Spin button every time.
Base Game	The primary screen of the game introduced for a play.
Bet	A sum of money staked. Betting is a sum of money risk against someone else's on the basis of the outcome of an unpredictable event (here - game).
Big Win	Is a game feature which appears over the reels with count up animation when win value is 40x initial bet or bigger.
Graphical User Interface (GUI)	Allows person to communicate with an electronic device through graphical icons and visual indicators, symbols.
Mock-up	Illustration of how the product should look like in different states.
Paytables	The list of payouts for a game. The table displays a list of winning combinations and the amount you are paid for each combination.
Quality Assurance (QA)	A systematic process of determining whether the software or product meets specified requirements.
QuickSpin	QuickSpin is an optional game state enabled by the Player, to reduce the duration of an ordinary spin.
Reels	The reels are what the player spins to make winning combinations. The reels are linked discs filled with pictures, or symbols, which relate to different kinds of wins.
Spin	Initiates a "reels" spin when a button is pressed.
Test Case	A test case is a document which consists of conditions or variables under which a tester determines whether the software satisfies requirements and functions correctly as expected.
Test Cycle (TC)	Final testing before game is officially signed off. Usually, there are 1-5 Test cycles.
Win Iterations	A display of an individual win.

Table of Contents

List of drawings

List of tables

1 Introduction

Nowadays games industry is in a great request and has rapidly grown in the development. Gambling has become increasingly popular and are in demand among the people. If we look deep in the games implementation process – testing tooks very important part of a game's successful development. In present tense testing covers vital part in all the software's development processes, regardless if it is games or other software production testing. Gambling industry has much competitors, consequently it is extremely important to be a distinguished partner on the gaming market. To achieve this respect companies should thoroughly and scrupulously approach to quality assurance of the producible software.

1.1 Background and problem

Derivco is an international software development company, which is dealing with online casino games design, development, testing and integrating solutions for one of the largest gambling providers – Microgaming Company, Derivco is one of the world's biggest suppliers of the online gaming. Every product must be tested and gone to certification before it goes life. Product testing is required after each product change or correction.

As the company develops online casino games, it is more difficult to test them through than other software production due to impossibility to test them automatically 100% effectively. It is necessary to cover the game different aspects with such tests, which will help to check everything maximally in the testing cycle. So far, manual testing is very important type of testing in software's quality assurance which helps to provide more qualitative product in the end result and guarantee a good experience of the final user.

1.2 The goal and tasks

The main goal of this work is to test a certain functionality of an online slot casino game developed by Derivco and afterwards when practical part of the game testing is complete to conclude how qualitative the product is on this stage, what are the most problematic areas that need more emphasis in the next testing cycle.

The goal of this work will be achieved by completing following tasks:

- Analyzing resources about manual testing and different testing methodologies.
- To figure out what is a game testing in general by analyzing different literature.
- To provide all the tested functional aspects in Derivco and choose the scope for this work's practical part.
- To analyze existing test cases and choose the most problematic zones what will be combined in scenarios for testing.
- To cover the game by functional tests of a chosen scope.
- To analyze which parts of the game need to be properly and profoundly checked in the next testing cycle and make conclusion about the quality of a tested product.

In the consequence of the work done, Author will gain more experience regarding testing practice and widen the knowledge. The circumspective testing plan and functionality will be maximally deliberated to provide functional scope's maximal test coverage. At the end of the practical part, Author will inform development team about found issues and testing results and the problematic areas to put more emphasis in the next testing cycle. In the conclusion of practical part, Author will make reccomendations about what testing methods in Derivco can be improved.

1.3 Methods used in this work

The main methodology used in this work is analysis. Author will analyze the literature about software testing and it's manual side, methods of testing. Author will compare the manual testing and automated testing including their pros and cons.

Author will perform analysis of a games testing process in Derivco and provide information about game's functional testing side and tested aspects.

One more methodology used in this work is description. The games testing part will take a place after the game expected functionality scenario's description of a chosen scope.

The final step after the theoretical side analysis and practical contribution to this work, on the basis of gained knowledge and obtained results Author will provide product quality and the most problematic functional areas analysis.

2 Company background

Derivco is an international software development company in the online gaming world established in 1996 in South Africa. Derivco designs and develops online casino games as well as integrates solutions for one of the largest gambling providers – for Microgaming Company.

Derivco was founded in Estonia in 2004 and mostly focused on developing of slot games for mobile and desktop platforms. [1] In addition to game design and development, Derivco gives a lot of attention to quality assurance, so that testing is really important part of game development process.

At this time, about 120 people work in Derivco Estonia dev house. In addition to South Africa and Estonia, the company has branches in Hong Kong, England, Isle of Man, Costa Rica, Australia and Sweden.

3 Definition of software testing

Software testing is a process of performing and exercising a program or application in order to find software defects and check it's quality and correctness. [2]

Testing can be defined as a process of validating and verifying that the software/product/application meets technical and business requirements, expectations and needs what were taken in account during its design and development and also if the product works as expected. [3]

3.1 The objectives of software testing

The main purpose of software testing is to gain confidence about the level of project quality. It is performed to verify that the ready software/project/product functionality corresponds to the expectations what are defined by the special project requirements or specifications. [4] The main goal is not to find as many product defects as you can, but it is important to uncover such situations, that could negatively impact on the usability of a product by the customer.

The software testing objectives and purposes are:

- To find defects which may get created while developing the software.
- To gain confidence and provide information about the level of quality of a software being delivered.
- To prove that the software has very few defects.
- To learn if the software is enough reliable.
- To ensure that the product meets business and user requirements.
- To get the trust and respect of customers by providing them a quality product.

[5]

Software testing is extremely necessary, because:

- Humans make mistakes during software development and some of them can turn expensive or dangerous. Is important to ensure that the software will not result into any failure.

- Only product with a good quality can be respected and have a good reputation in the market.

- Effective performance of a software application can be gained only by virtue of a good quality assurance.

Software testing is important to learn about the reliability of the software and application. [6]

3.2 Software testing methods and types

Software testing process can be done by using different testing methodology, such as:

- **Black-Box Testing** – functional or specification-based testing method, focuses on an output. Tester's coding knowledge is not necessary, there is no access to the source code. The tester is working at user interface level interacting with the system by providing inputs and enquiring outputs.

- **White-Box Testing** – testing method, that needs coding knowledge and detailed investigation of internal logic and code structure. When the software application fails, testers need to look inside the source code trying to find the cause and analyze which unit of code is behaving irrelevantly.

- **Grey-Box Testing** – application testing method, requires a limited knowledge of the internal operations of a product. The tester has access to design documents and the database, in contrast to black-box testers, who test only the product's user interface. Having some knowledge in internal mechanisms, tester can prepare better test scenarios/data in a test plan.

[7] [8]

Software testing process is divided into many testing types. Subsequently given types are most popular and well-known ones among the testing:

- Installation testing – conducted to ensure if the end-user can install and run the program properly on different operating systems.

- Usability testing – measures how well the GUI is designed and how easy it is in use.

- Exploratory testing – informal testing method, which is performed to explore the application and look for defects that exist in the application. Requires minimal planning and testers continuously make decision on the next step of action.

- Smoke testing – ensures whenever a new provided build of a software is without major issues and defects which will prevent the testing team to test properly.

- Regression testing – ensures that existing software's change, addition or any improvement has not broken or affected any existing functionality/features.

- Compatibility testing – checks how software behaves and runs on different environments, for example compatibility of a product with various systems – on different operating systems and web browsers.

- Localization testing – ensures how the software can be adapted to different languages and different market's demands.

- Back-end testing (database testing) – in this tests, testers have access to the database, they can easily connect and verify data by running few queries on that.

- Boundary value testing – checks the application's boundary level and whether defects exist in boundary values.

- Functional testing – focuses on the output to check if the product works as per the requirements.

- Graphical User Interface testing – validates if the GUI is as per expected design documents/mockups.

- Performance testing – checks if the system meets the performance requirements in different scenarios or not. [9]

4 Manual software testing

Software testing can be divided in two ways – manual testing and automation testing.

Manual testing process is carried out manually with the purpose to find defects or errors without using any automation tools. In this way of testing, tester plays an important role of the end user and checks that all the features of the product are working properly. [10] The main purpose of manual testing is to make it clear that the product under test is defect free and works as per required specification document. [11]

Manual testing process include following procedures:

- The analysis of requirements

  The requirements analysis process involves the determination of user expectations for a product. To get game requirements it is necessary to analyze functional specifications, game prototypes, animations, mock-ups and sound specifications.

- The creation of a test plan

  A special test plan describes the common testing approach, methods and the strategy being used during testing. The primary focus of a test plan is to give an overview about tasks and milestones of testing, define the shape and size of the test effort. Test plan is necessary to have full test coverage.

- The creation of a test cases

- The execution of test cases

- Logging defects in an issue tracking system, for example – Jira

  In a defect logging process tester should designate the issue's priority. The issue priority can be divided in:

    - Urgent – issue results in a critical impact on a business, making a production or it's part unusable.

- High – issue results in a high impact on a business affecting production and prevents some important functions from working properly.

- Medium – issue results in a medium impact on a business, where user is possible to experience partial non-critical functionality loss and the usability of a product is not affected significantly.

- Low – issue results in a minimal impact on a business which does not critically affect software's functions.

- The found defects fixing and their re-verification.

[12]

4.1 The advantages of manual testing over automated testing

Both types of testing - manual and automated offer advantages as well as disadvantages.

As mentioned before, in manual testing type – test cases are executed manually by the human, without any help of automation tools what contradicts to automated testing process, where test cases are executed with the assistance of different tools, scripts.

Testing has many specific methods available, some of them suit more to manual testing and some of them, conversely, best performed through automated tests. [13]

The reasons why manual testing is important enough and in demand:

- Manual testing is more beneficial if some test cases need to be checked a small number of times.

- Humans have more chances to find bugs and defects paying attention to details, no tool or machine can fit the power and capacity of human's brain and experience. Manual testers can quickly identify when something looks incorrect.

- The manual testing allows the tester to conduct additional special tests (ad-hoc testing).

- Manual testing can be performed by any tester, no need in a coach and special training given on automation tools.

- A good programming knowledge is required to write a script for an automation tool. In manual testing that kind of proficiencies are not required.

- Visual accessibility and preferences are really complicated to automate. Scripts don’t pick up some visual issues, thus Graphical User Interface testing can be done carefully and thoroughly with the help of manual testing.

- Human resources such as intuition, inference, logical conclusions, assumptions and reasoning can not be substituted by automation testing.

- If any accidental analysis is needed, automation testing can not be stopped for a while.

- Automated tests as well can consist of errors and holes what mean that scripts can return wrong false positives and false negatives. In this case a human touch is really significant.

[14] [15]

5 Game’s testing on a Derivco example

A game tester thoroughly tests games before the latest version is released to the public. The main purpose is to find all the defects in different game states with different scenarios in the final testing phase to make the game qualitative and prevent the game from being unplayable. The tester must try to do such steps within the game that an average player will not do, therefore, tester will know if the game behavior is correct and the game does not fail in extra particular states. Testers need to check lots of different platforms where the game can be played, depending on the location and company.

As the typical testing process the game testing also consist of:

- Requirements analysis. Tester should understand requirements as functional terms and conditions and analyze how the game components can be tested. All corresponding documentation should be read.

- In the sequel, testers start working on their testing documentation, including Test Plan and Test Cases development.

Regular and systematic game testing means that all the different parts and game features should be tested through.

The following is a list of areas and parts of the online casino slot game to be focused on during the application testing: Brand New User functionality, Game UI Screens, Animations, Bonus Feature functionality, Free Spins feature functionality, Achievements functionality, Big Win functionality, Settings Screen functionality, Bet Selector functionality, Autoplay Functionality, Menu Tab functionality, Boundary functionality, Modal Headers/Footers functionality, Credits functionality, Currency Support, Console functionality, Sounds functionality, Game closing functionality, Notification dialogs, Lobby integration.

Each game has a specific set of features.

5.1 Game functional testing side

The functional testing reviews every part of a piece of a software to make sure it operates and functions correctly in conformance with the requirement specification. Performing functional checks tester looks at what the software/application is supposed to do in some scenarios and makes sure if it actually does that.

The techniques used for functional testing are mostly based on special documentation and tester's experience. Requirements are analysed and prioritized at the early stage of the project development in different Functional Specifications.

To test game logic/functionality the QA specialist has to:

- Test all the game states – to ensure that game is not freezing at any state, to check different game states on re-enter.
- Check whole scenarios of the game.
- Ensure that all buttons are functioning in a correct way.
- Ensure that game is displaying an expected result in all the scenarios.

Following this method, if the expected results and received (actual) results match, then is possible to conclude that the software/application operates and proceeds properly and the functionality test has passed. On the contrary, if the expected and received results do not match, then it is quite clear there is a problem or some defect in the software's work. [16]

5.1.1 The non-functional testing of a game

The non-functional testing checks the performance, reliability, scalability and other non-functional aspects of the software system. As a rule, non-functional testing should be performed after functional testing. This type of testing circumscribes and characterizes how good the product works and behaves.

The non-functional testing includes compatibility tests to ensure that game is playable in all the game states on all supported devices and to ensure that performance is acceptable. Furthermore, non-functional testing includes stress tests to determine the minimum required Internet connection speed and time needed for game loading. [17]

6 Practical part. The online casino game functional testing

According to software's testing types description in 3.2 section numbers there are many diverse testing types in a testing conception. In this work, Author will perform functional testing in a final testing phase before project sign-off. Final testing takes place after dev QA specialists (testers in development teams) have been tested the game. Final testing will be performed in the first testing cycle (usually there are up to 5 testing cycles for each project). The purpose is to check whether the game conforms with all the requirements. Functional testing will be performed through Black-Box testing method - Author will perform specification-based testing and will not access the source code. Author will concentrate on functional testing as it is a wide technology and can be called a most important part of software testing which is primarily used to verify that a piece of a product is providing the same output as required by the end-user/business. Furthermore, functional testing gives opportunity to check the software for usability to ensure that the navigational functions are working as required. The choice to the functional testing side was made because of it's importance for every product quality assurance and it's wide product coverage. The functional testing will be done through exploratory testing. In addition to functional testing, Author will perform usability testing type.

As well as functional testing type this work will include GUI testing which is considered to be done simultaneously with functional testing. Author examines this type of testing because the user interface of a product is a critical part of the software's development lifecycle. A poorly designed user interface undoubtedly leads to dissatisfied customers. Consequently, it is undeniably necessary to prevent as many defects as possible from reaching the UI level.

In a practical part of this work Author will take the online casino slot game called "Dream Date" developed and implemented by Derivco and will perform functional testing of that game, including GUI testing and usability testing methods. The functional testing will be performed on a Desktop version of the game.

Primarily, Author will deal with test plan with description of parts what will be taken in account during functional game testing. In the next steps Author will execute testing scenarios for "Dream Date" functionality ensure the game meets expectations in its

functional side. All the testing process will be done with help and support of special tools necessary for testing implementation.

6.1 About „Dream Date“ game

Dream Date is a video slot game with Modern and Historical Romance theme. It has 5x3 Spinning Reels, 243 Number of Ways. This game has both Wild and Scatter symbols. It is possible to change an era in this game from Modern to Historic and vice versa, also this game offers a possibility to change your „Dream Date”. There is Hot Zones feature in a base game, what awards extra Cash prize. Also, Dream Date has Free Spins feature with expanding wild functionality.

Following picture shows how the game looks like in a practice:

Picture 1. Dream Date Base Game Screen (historic)

6.2 Test planning

As it was said above in the section 5, every game has a specific set of features. Information about game features is received from the Functional Specification and each game has wide functional coverage. Testing each functional part of a game is very laborious and time-consuming process that needs much attentiveness and diligence.

Author will focus on following areas during functional testing process:

- Big Win functionality
- Achievements functionality

- Settings Screen functionality
- Menu Tab functionality
- Autoplay functionality

These parts are the testing scope of current work because of their importance. These parts are the main part of the online casino game's functionality and an integral part of each game. The check would consist of selected scenarios from the existed ones. Author has made an analysis and filtration of existent test cases to choose the most problematic scenarios what mean that those scenarios are more likely to have defects executing them. Furthermore, Author combined different game scenarios in one to provide more compactness in the granted functionality information as well as to provide the most effectively covered game functionality checks. Those scenarios consist of more than one test case in each. For the issues found, Author will provide description of the defect, defect priority and explanation about incorrectness. For some issues Author will attach screenshots (if it will be a possibility and appropriate case).

6.2.1 Big Win functionality testing

Subsequently, Author will choose and review such scenarios of a Big Win functional side, which are more likely to meet defects.

The following is a consolidated table of Big Win Scenarios and their priorities:

Area:	**Big Win Functionality**	
Scenario Nr	**Priority**	**Scenario short description**
1	Medium	Big Win trigger and it's tiers check
2	Medium	Corresponding sound should be played during each Big Win tier
3	High	Big Win removal behavior by two-click interaction
4	Low	Big Win display timings
5	High	Big Win behavior with menu tab
6	High	Big Win count up must be in sync with console win box count up
7	Medium	Disconnection from the game during Big Win count up
8	Low	Resizing the browser window during Big Win

Table 1. Big Win scenario descriptions and their priorities.

Scenario 1: Big Win trigger and it's tiers check

Steps:

1. Trigger Mega Big Win
2. observe it's behavior during tiers increase

Expected result according to specification: Big Win should appear on trigger. Big Win has 3 tiers (Big Win, Super Big Win, Mega Big Win). Count up starts from Big Win banner, Big Win graphics should update for each tier accordingly:

- Big Win ≥ 40x total bet

- Super Big Win ≥ 100x total bet

- Mega Big Win ≥ 250x total bet

As the next Tier is reached, the Big Win text must animate to the next Tier, the density of hearts and fireworks animation must increase.

Received result of a check: Author has examined appropriate scenario, and the check result was in accordance to expected result - scenario has passed.

Scenario 2: Corresponding sound should be played during each Big Win tier

Steps:

1. Trigger Mega Big Win
2. Listen to the sounds played when transition to the next tier happens

Expected result according to specification: corresponding Win tier and transition sound should play for Super/Mega Big Win tiers until count up is completed or interrupted. Sound should play at the exact time the Big Win banner skips to another tier.

Received result of a check: Author has examined appropriate scenario and the check result was not in accordance to expected result what means that this scenario contains issues in its functionality. The scenario has failed.

Issue description: ***Tier transition sound is played after visual tier transition already happened, what contradicts to expected functionality behavior circumscribed above.***

In the received result tier transition sound is played after Big Win banner tiers transition happens. Consequently, it is the irregular behavior of the Big Win functionality.

Scenario 3: Big Win Removal behavior by two-click interaction

Steps:

1. Trigger Big Win
2. Click on the screen during count up
3. Click again when count up is interrupted

Expected result according to specification*:* on the first click count up must summarize in the Big Win box. On the second click Big Win banner must fade out over 0.5 sec.

Received result of a check: Author has examined appropriate scenario, and the check result was in accordance to expected result - scenario has passed.

Scenario 4: Big Win display timings

Steps:

1. Trigger Big Win
2. Observe Big Win banner fade in, display and fade out timings

Expected result according to specification: Big Win banner should fade in over 0.5 seconds and display for 2 seconds after count up complete before the banner removal. Afterwards, it should fade out over 0.5 seconds.

Received result of a check: Author has examined appropriate scenario and the check result was not in accordance to expected result what means that this scenario contains issues in its functionality. The scenario has failed.

Issue description: ***Big Win box display timing is incorrect, what contradicts to expected functionality behavior circumscribed above.***

In the received result the Big Win fade in/out timings are correct, but the banner display timing after count up complete is wrong – it was displayed for 3 seconds, what is unexpected behavior of its functionality.

Scenario 5: Big Win behavior with menu tab

Steps:

1. Trigger Big Win
2. Open menu tab during Big Win count up
3. Navigate to Paytables
4. Navigate back to the Base Game

Expected result according to specification*:* the selected element (here: Paytables) must slide out and Big Win banner should still be displayed during that time. When user navigates back to the Base game, the Big Win banner must be displayed.

Received result of a check: Author has examined appropriate scenario and the check result was in accordance to expected result - scenario has passed.

Scenario 6: Big Win count up must be in sync with console win box count up

Steps:

1. Trigger Big Win
2. Observe Big Win count up in the Big Win box and console win box

Expected result according to specification: Big Win count up must be in sync with the win count up in the console win box. After the count up has finished the total win displayed must be the same as in the Big Win box.

Received result of a check: Author has examined appropriate scenario and the check result was in accordance to expected result - scenario has passed.

Scenario 7: Disconnection from the game during Big Win count up

Steps:

1. Trigger Big Win
2. Re-enter to the game
3. Observe Big Win behavior

Expected result according to specification: Big Win should not be presented on re-enter to the game. Win value should be displayed in the win box and win iterations should stay playing.

Received result of a check: Author has examined appropriate scenario and the check result was in accordance to expected result - scenario has passed.

Scenario 8: Resizing the browser window during Big Win

Steps:

1. Trigger Big Win
2. Resize the browser window to different resolutions

Expected result according to specification: Big Win banner should adjust smoothly according to the screen resolution changes. Graphics should not break.

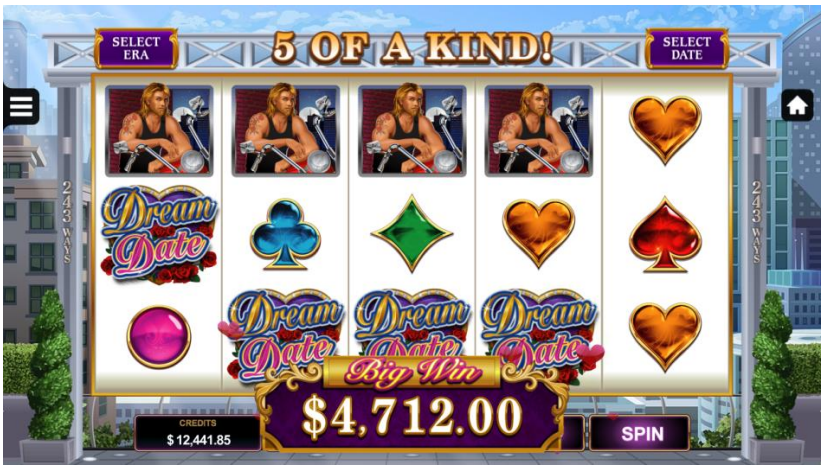
Received result of a check: Author has examined appropriate scenario and the check result was not in accordance to expected result what means that this scenario contains issues in it's functionality. The scenario has failed.

Issue description: ***BigWin Banner is misplaced after resizing the window, what contradicts to expected functionality behavior circumscribed above.***

In the received result Big Win banner is misplaced. Following picture shows the difference from approved Big Win banner position mock-up.

Picture 2. Defect 1. Big Win banner incorrect placement

Picture 3 shows the approved mock-up of the Big Win banner correct placement.



Picture 3. Approved mock-up of the Big Win correct placement

Conclusion:

During the check of Big Win scenarios below, Author has found three issues related to Big Win functionality incorrectness. Following table shows Big Win scenarios and their check results.

Scenario Nr	Description	Status
1	Big Win trigger and it's tiers check	Passed
2	Corresponding sound should be played during each Big Win tier	Failed
3	Big Win removal behavior by two-click interaction	Passed
4	Big Win display timings	Failed
5	Big Win behavior with menu tab	Passed
6	Big Win count up must be in sync with console win box count up	Passed
7	Disconnection from the game during Big Win count up	Passed
8	Resizing the browser window during Big Win	Failed

Table 2. Big Win Scenarios and check results

According to the issues found, there are two issues regarding Big Win functionality and one graphical issue what refers to GUI issue type.

6.2.2 Achievements functionality testing

Subsequently, Author will choose and review such scenarios of achievements functional side, which are more likely to meet defects or bugs. The following is a consolidated table of Achievements Scenarios and their priorities:

Area:		Achievements Functionality
Scenario Nr	**Priority**	**Description**
1	Low	Achievement plaque graphics and alignment
2	High	Achievement plaque functionality
3	Low	Achievement plaque display timings
4	Medium	Achievement plaque behavior with menu tab
5	High	Achievement plaque behavior with Big Win
6	Medium	Achievement plaque behavior and timings with QuickSpin mode on

Table 3. Achievements scenario descriptions and their priorities

Scenario 1: Achievement plaque graphics and alignment

Steps:

1. Get an achievement unlocked
2. Observe graphics

Expected result according to specification: achievement plaque should be located correctly, the text and symbol image should not be truncated or misplaced.

Received result of a check: Author has examined appropriate scenario and the check result was not in accordance to expected result what means that this scenario contains issues in its functionality. The scenario has failed.

Issue description: ***Symbol on achievement plaque is shifted, what contradicts to expected functionality behavior circumscribed above.***

Following picture shows the misplaced symbol on achievement plaque, what contradicts to the game correct graphics.

Picture 4. Defect 1. Symbol on achievement plaque is misplaced

Picture 5 shows the correct placement of an achievement plaque:

Picture 5. Approved mock-up of an achievement placement

Scenario 2: Achievement plaque functionality

Steps:

1. Get an achievement unlocked
2. Observe it's behavior, try to click on the achievement plaque
3. Refresh the game

Expected result according to specification: achievement plaque should be displayed over the reels if all possible win combinations for a particular symbol have appeared. It should not be clickable and can not be dismissed. After re-enter to the game, achievement plaque should not be displayed.

Received result of a check: Author has examined appropriate scenario and the check result was in accordance to expected result - scenario has passed.

Scenario Nr 3: Achievement plaque display timings

Steps:

1. Get an achievement unlocked
2. Observe display timing
3. Repeat this step one more time, but do not wait until it disappears by itself, initiate a spin earlier
4. Get achievement unlocked during Autoplay

Expected result according to specification: achievement plaque should fade in over 0.5 seconds and then display for 2 seconds before 0.5 sec fading out. The same timings stay

during Autoplay. The achievement plaque should fade out immediately, if player initiates spin before the 2 seconds end.

Received result of a check: Author has examined appropriate scenario and the check result was in accordance to expected result - scenario has passed.

Scenario 4: Achievement plaque behavior with menu tab

Steps:

1. Get an achievement unlocked
2. Click on the menu tab

Expected result according to specification: while menu is opened achievement plaque should be displayed behind the menu for standard amount of time, which is equal to 2 seconds.

Received result of a check: Author has examined appropriate scenario and the check result was in accordance to expected result - scenario has passed.

Scenario 5: Achievement plaque behavior with Big Win

Steps:

1. Trigger Big Win
2. Get any achievement unlocked at the same time

Expected result according to specification: the achievement standard functionality should not break as well as Big Win functionality. Achievement plaque should layer above the reels as it was stated above.

Received result of a check: Author has examined appropriate scenario and the check result was in accordance to expected result - scenario has passed.

Scenario 6: Achievement plaque behavior and timings with QuickSpin mode on

Steps:

1. Turn QuickSpin option on in Settings
2. Get an achievement unlocked

Expected result according to specification: the achievement plaque should fade in immediately as the next spin starts without any delay and hold for 2 seconds.

Received result of a check: Author has examined appropriate scenario and the check result was not in accordance to expected result what means that this scenario contains issues in its functionality. The scenario has failed.

Issue description: ***Achievement plague is displayed on the screen for ~2 sec after Spin is initiated with QuickSpin mode on, what contradicts to expected functionality behavior circumscribed above.***

In the received result achievement plague remains on the screen for 2 secs, the next spin does not start until achievement plaque disappears.

Conclusion:

During the achievements functionality check, Author has found two issues related to achievements functionality behavior incorrectness. Following table shows Achievements scenarios and their check results.

Scenario Nr	Description	Status
1	Achievement plaque graphics and alignment	Failed
2	Achievement plaque functionality	Passed
3	Achievement plaque display timings	Passed
4	Achievement plaque behavior with menu tab	Passed
5	Achievement plaque behavior with Big Win	Passed
6	Achievement plaque behavior and timings with QuickSpin mode on	Failed

Table 4. Achievements scenarios and check results

According to the issues found, there is one issue regarding Achievements functionality and one graphical issue what refers to GUI issue type.

6.2.3 Settings Screen functionality

Subsequently, Author will choose and review such scenarios of Settings functional side, which are more likely to meet defects or bugs. The following is a consolidated table of Settings Scenarios and their priorities:

Area:		Settings Screen Functionality
Scenario Nr	**Priority**	**Description**
1	High	Settings Screen availability
2	High	Settings Screen layout
3	High	Opening Settings Screen should not affect game behavior
4	Medium	Check Sounds and QuickSpin switch functionality

Table 5. Settings Screen scenario descriptions and their priorities

Scenario 1: Settings Screen availability

Steps:

1. Navigate to menu
2. Open Settings Screen

Expected result according to specification: setting screen should open. Return arrow should be displayed in place of menu tab and after pressing that user is returned to menu tab.

Received result of a check: Author has examined appropriate scenario and the check result was in accordance to expected result - scenario has passed.

Scenario 2: Settings Screen layout

Steps:

1. Navigate to menu
2. Open Settings Screen

Expected result according to specification: Settings Screen is divided in two areas: the top container on the top of the screen and the middle container on the centre of the remaining screen space. The middle container consist of: Coins Size, Coins Bet drop down menus, Total Bet box (should show actual bet), Sounds and QuickSpin switches. The “Powered by Microgaming TM” text and “Responsible Gaming” link should be displayed at the bottom part of the screen.

Received result of a check: Author has examined appropriate scenario and the check result was in accordance to expected result - scenario has passed.

Scenario 3: Opening Settings Screen should not affect game behavior

Steps:

1. Get any win combination in the base game
2. Navigate to menu
3. Open Settings Screen
4. Return back to the base game

Expected result according to specification: win results and animations should be still displayed, win count up should be interrupted and total win value should be displayed in the win box.

Received result of a check: Author has examined appropriate scenario and the check result was in accordance to expected result - scenario has passed.

Scenario 4: Check Sounds and QuickSpin switch functionality

Steps:

1. Navigate to menu
2. Open Settings Screen
3. Switch Sounds and QuickSpin on and off
4. Observe the Base Game behavior.

Expected result according to specification: If QuickSpin and Sounds are not enabled in the Base Game, then „QuickSpin“ and „Sounds“ switches are OFF in the Settings Screen. If „QuickSpin“ mode is enabled, reels stop quickly. On the game refresh, the last selected states of switches should stay in the Settings Screen.

Received result of a check: Author has examined appropriate scenario and the check result was not in accordance to expected result what means that this scenario contains issues in its functionality. The scenario has failed.

Issue description: QuickSpin toggle state does not save on refresh, what contradicts to expected functionality behavior circumscribed above.

In the received result QuickSpin toggle in the Settings Screen does not save last condition when refresh is made. According to specification, the last selected state should be saved.

Conclusion:

During the Settings Screen functionality check, Author has found one issue regarding it's functionality. Following table shows Settings Screen's functionality scenarios and their check results.

Scenario Nr	Description	Status
1	Settings Screen availability	Passed
2	Settings Screen layout	Passed
3	Opening Settings Screen should not affect game behavior	Passed
4	Check Sounds and QuickSpin switch functionality	Failed

Table 6. Settings Screen scenarios and check results

According to the table there is one issue regarding Settings Screen functionality.

6.2.4 Menu Tab functionality

Subsequently, Author will choose and review such scenarios of Menu functional side, which are more likely to meet defects or bugs. The following is a consolidated table of Menu Scenarios and their priorities:

Area:		Menu Functionality
Scenario Nr	Priority	Description
1	High	Menu open and close functionality
2	Medium	Menu items and graphical check
3	High	Menu behavior with spin and bet button
4	High	Menu items behavior on selection

Table 7. Menu tab scenario descriptions and their priorities

Scenario 1: Menu open and close functionality

Steps:

1. Open menu
2. Close that by clicking the return arrow or anywhere on the game screen (except menu area)

Expected result according to specification: if player opens menu, it slides in from the left hand side of the screen over 0.4 seconds. The Menu always layers above every other game element. The menu container utilizes a darkened alpha layer (80%) so that the game background can be visible behind. Menu tab closes over 0.2 seconds and game should not be affected. No sounds playing when accessing menu tab and closing it.

Received result of a check: Author has examined appropriate scenario and the check result was in accordance to expected result - scenario has passed.

Scenario 2: Menu items and graphical check

Steps:

1. Open menu
2. Observe menu and it's items

Expected result according to specification: the following options should be present on menu tab: banking, settings, paytables. Each menu link should be represented in text with its unique icon displayed on the left. Horizontal rules must appear between each menu item.

Received result of a check: Author has examined appropriate scenario and the check result was in accordance to expected result - scenario has passed.

Scenario 3: Menu behavior with spin and bet button

Steps:

1. Open menu
2. Try to access spin and bet button which is placed under opened menu

Expected result according to specification: when player clicks on spin button the menu tab closes, but the spin is not initiated. When player tries to access bet button menu remains opened, what means that bet button is disabled.

Received result of a check: Author has examined appropriate scenario and the check result was in accordance to expected result - scenario has passed.

Scenario 4: Menu items behavior on selection

Steps:

1. Open menu and tap on any menu item

Expected result according to specification: press area for menu links should be bigger than text and follow line dividers. Menu item text should change to selected state and remain until the relevant menu element has slid out. Unselected links should be

functionally disable but remain visually enabled. Menu should remain opened during the item slide out. No sounds should play during this functionality.

Received result of a check: Author has examined appropriate scenario and the check result was in accordance to expected result - scenario has passed.

Conclusion:

During the Menu Tab functionality check no associated issues were found in the functionality behavior. All scenarios have passed checks.

6.2.5 Autoplay Functionality

Subsequently, Author will choose and review such scenarios of Autoplay functional side, which are more likely to meet defects or bugs. The following is a consolidated table of Autoplay Scenarios and their priorities:

Area:		Autoplay Functionality
Scenario Nr	**Priority**	**Description**
1	High	Autoplay button behavior
2	High	Autoplay Spin Counter Box and Stop Autoplay button behavior
3	High	Start Autoplay and it's functionality
4	High	Disconnect during Autoplay

Table 8. Autoplay scenario descriptions and their priorities

Scenario 1: Autoplay button behavior

Steps:

1. Press on Autoplay button
2. Close it by clicking anywhere on the screen
3. Open panel one more time and choose any of preset values.

Expected result according to specification: the Autoplay Pop Up Panel must immediately slide up, the panel should consist of 10x, 25x, 50x, 100x, „Until Stop", „Custom" value options. Panel must immediately be closed (slide down) by clicking anywhere on the screen including console element. On selecting of any preset value the Pop Up Panel must be removed, the SPIN button must be replaced with a STOP AUTOPLAY button, the Autoplay button must be replaced with a Spin Counter Box, the Autoplay session must begin.

Received result of a check: Author has examined appropriate scenario and the check result was in accordance to expected result - scenario has passed.

Scenario 2: Autoplay Spin Counter Box and Stop Autoplay button behavior

Steps:

1. Start Autoplay
2. Observe Spin Counter Box
3. Wait until some spins complete
4. Click on Stop Autoplay button

Expected result according to specification: Spin Counter should have x / y layout, where the left value begins at 1 and count up at the start of each spin. The right value reflects the total spins to be played in the current Autoplay session. When stop Autoplay button is pressed, once the Reels have resolved, before the next spin begins the Spin Counter Box must immediately change to an active AUTOPLAY button.

Received result of a check: Author has examined appropriate scenario and the check result was in accordance to expected result - scenario has passed.

Scenario 3: Start Autoplay and it's functionality

Steps:

1. Start Autoplay by choosing 10x preset value
2. Observe the game behavior

Expected result according to specification: Autoplay with 10 spins should start without any delay, after 10 spins complete Autoplay should stop. The console must remain disabled throghout the Autoplay session. The Menu tab and Stop Autoplay buttons must remain enabled.

Received result of a check: Author has examined appropriate scenario and the check result was in accordance to expected result - scenario has passed.

Scenario 4: Disconnect during Autoplay

Steps:

1. Start Autoplay

2. Refresh the game page

Expected result according to specification: Autoplay is canceled after refresh. Spin and Autoplay buttons are present without any count down numbers. Last spin combination should be present on reels.

Received result of a check: Author has examined appropriate scenario and the check result was in accordance to expected result - scenario has passed.

Conclusion:

During the Autoplay functionality check no associated issues were found in the functionality behavior. All scenarios have passed checks.

6.3 The analysis of obtained results

During the Dream Date online casino game functional testing in the final testing phase first test cycle Author received such results:

- Big Win functionality check was covered by 8 different scenarios in which 3 issues were found.
- Achievements functionality check was covered by 6 different scenarios in which 2 issues were found.
- Settings Screen functionality check was covered by 4 different scenarios in which 1 issue were found.
- Menu Tab and Autoplay functionality checks were covered each by 4 different scenarios and no issues were found during those checks.

Totally, there was 6 issue found, 4 of them were functional issues and 2 of them – graphical issues. Functional issues included timings, sounds and disconnection scenario issues.

On the assumption of received results of the Dream Date online casino game functional side testing is possible to conclude that in a greater degree game meets expectations and requirements.

On the analysis of found issues Author can make deduction that some functionality parts as, for example, Big Win and Achievements functionality need to be checked more properly because of their complexity and elaboration – they are most problematic parts from the scope and the found issues percentage in these areas proves that. Two issues were regarding a game functionality timings, what mean that such areas as Timings for different game features should be tested with more effort and advertency because they are considered as painstaking game areas and need more concentration. It is also important to put more emphasis on sounds part, because it is really difficult to take cognizance of some unnoticeable little defects. Really important is to check such conditions in which game goes out of the comfort zone, for example to test not supported resolutions and the different scalings of the browser window. Game testing process should include disconnection scenarios tests because they can be problematic and it is really important to simulate different game scenarios done by the player to provide a satisfying product quality. Likewise, some issues can be found because of code correction by developers and their inadvertency. Those issues are just unnoticed and should be found by attentive QA specialists.

To recapitulate, the most problematic parts to put more emphasis in the next testing cycles are Big Win and Achievements parts. Testers should put more attention to timings, sounds and unsupported resolutions.

To look on the usability side, Author can conclude that the game user interface is intuitive and comfortably designed. The Dream Date is easy to use game, players can reach their goals without a long training. The GUI is well designed and quite understandable.

6.4 Possible improvements in games testing process in Derivco

In the sequel of the work done, some under test scenarios included the sounds requirements and their testing was quite time consuming. In the parts different scenarios check Author has found 1 issue related to game sounds. Author can suggest improving Sounds testing process in Derivco Quality Assurance and make this side of tests more effective and easy.

Usually sounds check of all the game's features takes 1 working day for 1 tester what is approximately 8 working hours. Green Lantern is an internal tool used by Derivco for

analyzing game Components/HID/Sounds, what affords the list of all available sounds and gives a possibility to play it through. This tool highlights playing sounds and fading periods for looping sounds. This tool is used only for a Desktop checks. All the other mobile devices checks are done by the audition and attentiveness of a QA specialist, what means that no other tools are used in help for Sounds testing process for mobile devices.

The Sounds testing process in Derivco can be simplified by virtue of GoldWave, RecForge Pro and Screen Recording tools.

GoldWave is a tool that is fully loaded to do everything from the simplest recording and editing sounds to the most refined audio processing, restoration, enhancements, analysis and conversions. GoldWave tool offers different real-time graphics visuals, what can provide more qualitative sounds check. It's wide functionality and easy to learn interface will provide fast and qualitative sounds check opportunity to the testing process. GoldWave tool can be used on a Desktop version specific checks.

To check sounds on Android devices is possible to use RecForge Pro tool, what has good sound quality and gives an opportunity to record the sounds from the game on android devices. In addition, it is possible to use Screen Recording tool on 11 iOS devices to record the game scenarios and their sounds. As was mentioned before, testers use their auditory abilities to check the sounds scenarios on mobile, what is significantly time-consuming practice. It will be noticeably easier to check mobile sounds scenarios after recording them, thus tester will not need to repeat different scenarios to check different sounds for a lot of times. It will be fast enough to check sounds by listening to recorded scenarios. Therefore, the time calculated on a sounds check on mobile devices will be shortened almost twice.

Consequently, the time spent on Sounds testing will be saved, fewer resources will be disbursed.

7 Conclusion

The main goal of this work set by the Author to test the online casino game functionality in a final testing phase and to make derivation about the explored product's quality and its problematic areas is achieved. The goal was successfully obtained by completing different tasks such as literature analysis, the game's functional side scope selection and the optimization of chosen game functionality in different scenarios. After the practical part Author has made recommendations to improve the testing process in Derivco.

According to this work is possible to conclude that manual testing side is really important - the human eye and concentration makes sense in product testing. Some functionality aspects testing is sufficiently time consuming, but in order to attain the required quality of a product time spent on manual testing worth it.

As a result of this work, Author gained new experience and increased knowledge in online casino games testing specific. This work demonstrates the successful possibility to make exploratory functional testing more effective by performing deep work with different scenarios and their combination. In consequence of the work done, Author have apprised development studio about found issues and problematic game areas. By virtue of the work done, more emphasis will be concentrated on these problematic areas in the future testing cycle and all functionality related issues will be examined and corrected. Furthermore, sounds functionality testing improvement suggestions will be taken into account to optimize and decrease time and resources spent for testing activities.

Used literature

[1] Derivco Estonia OÜ, "Tallinn, Estonia," [Online]. Available: https://derivco.com/locations/estonia/. [Accessed 14 03 2018].

[2] L. Sharma, "Software Testing," 9 04 2016. [Online]. Available: http://toolsqa.com/software-testing/software-testing/. [Accessed 14 03 2018].

[3] I. C. Tester, "What is software testing?," [Online]. Available: http://istqbexamcertification.com/what-is-software-testing/. [Accessed 14 03 2018].

[4] RightHand Technologies, "Software Testing," [Online]. Available: http://www.righthandtech.com/software-testing.php. [Accessed 14 03 2018].

[5] I. C. Tester, "What is software testing objectives and purpose?," [Online]. Available: http://istqbexamcertification.com/what-is-the-software-testing-objectives-and-purpose/. [Accessed 14 03 2018].

[6] I. C. Tester, "Why is software testing necessary?," [Online]. Available: http://istqbexamcertification.com/why-is-testing-necessary/. [Accessed 17 03 2018].

[7] L. Bradford, "Everything You Need to Know About Software Testing Methods," 08 05 2017. [Online]. Available: https://www.thebalancecareers.com/all-you-need-to-know-about-software-testing-methods-4019921. [Accessed 17 03 2018].

[8] Tutorials Point, "Software Testing - Methods," [Online]. Available: https://www.tutorialspoint.com/software_testing/software_testing_methods.htm. [Accessed 17 03 2018].

[9] Software Testing Help, "Types of Software Testing," 8 11 2017. [Online]. Available: https://www.softwaretestinghelp.com/types-of-software-testing/.

[10] L. Sharma, "Manual Testing," 9 04 2016. [Online]. Available: http://toolsqa.com/software-testing/manual-testing/. [Accessed 18 03 2018].

[11] Software Testing Class, "What is Manual Testing?," [Online]. Available: http://www.softwaretestingclass.com/what-is-manual-testing/. [Accessed 17 03 2018].

[12] JumpMind, "Priority Levels," [Online]. Available: http://www.jumpmind.com/services/support/priority-levels. [Accessed 03 05 2018].

[13] Apica, "Automated Testing vs Manual Testing: Which Should You Use, and When?," 7 11 2014. [Online]. Available: https://www.apicasystems.com/blog/automated-testing-vs-manual-testing/ . [Accessed 07 04 2018].

[14] P. Sparrow, "Manual Testing: Advantages and Disadvantages of Manual Software Testing," [Online]. Available: https://www.ianswer4u.com/2016/12/manual-testing-advantages-and.html. [Accessed 07 04 2018].

[15] T. Wyher, "5 Reasons Why Manual Testing Is Still Very Important," 26 10 2016. [Online]. Available: https://dzone.com/articles/5-reasons-why-manual-testing-is-still-very-importa. [Accessed 07 04 2018].

[16] QASymphony, "Functional Testing Types – 25 Best Practices, Tips & More!," [Online]. Available: https://www.qasymphony.com/blog/functional-testing-types/. [Accessed 07 04 2018].

[17] Guru99, "FUNCTIONAL Testing Tutorial: What is, Process, Types, & Examples," [Online]. Available: https://www.guru99.com/functional-testing.html. [Accessed 07 04 2018].