



Quality Control in Games Development

Marwa Syed

2019 Laurea



Laurea University of Applied Sciences

Importance of Quality Control in Games Development

Marwa Syed
Business Information Technology
Bachelor's Thesis
May 2019

Marwa Syed

Development

Year	2019	Pages	45
------	------	-------	----

The purpose of this thesis was to provide more information regarding the role of quality control in game development to game industry professionals, game development start-ups and potential employees of this field. There are several misconceptions about this profession and this thesis contains information regarding the impact quality control has in games development.

The motivation to conduct this research and write this thesis comes from the author's work experience in testing games in a game development studio. From personal experience, the author had some misconceptions about this profession as well, which were cleared when she began working as a tester. The information regarding quality control is sparse compared to other aspects of game development.

This thesis was written using a qualitative research method where the sources were from data collection, online survey, printed and online literature. The results of this research outlined the different methods of testing processes the quality control department must undergo to produce a feasible outcome for a game.

Quality control is one of the main aspects of game development, since this activity results in the discovery of bugs, establishing a defect database, and most importantly, the approval for a viable game to be published. Quality control does not cease when the game is released but continues postproduction as well.

The conclusion of this thesis report is that quality control is vital to the development of a game, although some game companies still do not view it as an important aspect for their games. In addition to hopefully clearing the misconceptions of people who want to follow this career path due to misunderstanding of the skills and knowledge required to be a games tester. Furthermore, an attempt to settle the mistaken belief that quality control is not a viable career path and people beginning at this position to gain a position in other departments of game development.

The author has signed a non-disclosure agreement with her employer. All information in this thesis that has been gathered from survey responses are taken from game industry professionals and are respectively referenced. Survey questions asked their opinions on quality control and when they feel testing is required most, whether quality control is required in start-ups, automated testing, and testers responsibility in detecting fun factor of the game.

Keywords: Game development, Testing, Quality Control

Table of Contents

1	Games Development Terminology	6
2	What is Quality Control?.....	8
2.1	Skills required to be Quality Control	8
2.2	Why is Testing Required?	10
3	Game Development	13
3.1	Different Roles in Game Development	14
3.1.1	Producer	14
3.1.2	Designer.....	15
3.1.3	Programmer	17
3.1.4	Artists	18
3.1.5	Audio	19
3.1.6	Quality Control	20
4	Game Development Process and role of QC	21
4.1	Concept	21
4.2	Pre-Production	22
4.3	Prototype	22
4.4	Production.....	22
4.5	Alpha	23
4.6	Beta	24
4.6.1	Open and Closed Beta testing.....	24
4.7	Gold	25
4.8	Postproduction	25
4.9	QC requirement in Start Ups.....	26
5	Quality Control in Depth	27
5.1	QC Manager	27
5.2	QC Lead.....	28
5.3	Compatibility Testers	28
5.4	Quality Assurance and Regression testers	28
5.5	Localization Testers	28
6	QC Processes	29
6.1.1	Automated Testing.....	33
6.1.2	Functionality Testing	34
6.1.3	Progression Testing	35
6.1.5	Regression Testing	35

6.1.6	Ad-hoc Testing.....	35
6.2	Tools used by QC.....	35
6.2.5	Bug recording tools	36
6.3	Types of Issues.....	37
6.3.1	Visual bugs	37
6.3.2	Audio bugs	379
6.3.3	Level Design	379
6.3.4	Issues with Physics.....	40
6.3.5	Stability Issues	40
6.3.6	Network Bugs	40
6.3.7	Monetization Issues.....	41
6.4	Bug Prioritization	41
6.5	Documentation	43
7	Research Methods	44
8	Conclusion	45

1 Games Development Terminology

AAA - A game that requires high-budget and a large company to be developed. These games are commonly available on several platforms and are highly promoted/marketed.

Alpha Release - An incomplete but functioning version of the game released to test/gain feedback about the main functionality and design of the game.

AoE - Area of Effect, is an effect of ability in games that affects several targets in a specific area. For e.g.: an explosion.

Balance - The fairness of a game depending on the relation between a character's abilities, i.e.: power, health, speed etc. It could also mean the fairness in how a player can level up in a game. For e.g.: a game can be pay to play (look at pay to play).

Beta Release - A completed version of the game released prior to commercial release to identify and resolve any remaining bugs/issues.

Closed Beta - Similar to Beta version of the game but only released to specific number of people.

CNR - Cannot reproduce

Collision - When two objects within a game have an impact with each other or can crash into each other.

Co-op - Also know. as Cooperative, describes multiplayer games where players can play with other players or friends against other players or computer AI.

Content Rating - This is how a game is classified according to its content, depending on several factors such as violence, sexual content, crude language etc. Depending on the country this rating will differ.

Crash - When the game unexpectedly shuts down without the permission of the user.

Cross Platform - When a game is available on several different platforms/consoles and can be cooperatively played regardless of which platform one user is playing the game on.

DLC - Downloadable Content, is additional content that a player can download for the game. Content is either free or can be purchased.

F2P - Free-to-Play, is a free game that does not require purchasing from the developer but has content within the game that can be purchased, e.g.: customizable items, items needed

to level up character, etc. These items can be gained through loot boxes (see loot box) or through progression but players can choose to spend money and purchase them instead.

Game Design - It is the method of designing the content and the rules of the game as well as what the game is going to be about. The basic idea of the game is created by game designers. There different types of designers such as level designers, content designers, game writers, technical designers etc.

Game Localization - This is translation of dialogues and text within the game for users from other countries. Some content could also be changed to suit the specific country as well.

HUD - Heads up Display, is the display area in game where user can see their characters statistics such as health and attributes.

IAP - In-App Purchases, is content that can be purchased within a mobile game that is usually free. Free to Play games apply this logic to gain revenue.

Loot Box - These are awards given to a player for gaining levels, completing challenges or any other achievement. The boxes contain level up items, cosmetic items etc. These loot boxes can also be purchased with real currency or in-game currency. Loot Boxes can also have other names such as packs or crates depending on the game.

NAD - Not a Defect

Patch - An update created for the game that consists of fixes and other changes made to the game.

PC- Personal Computer, which is a platform in which games can be played

QC - Quality Control

Sandbox - This is a virtual environment where the game can be run and tested securely. There can be one environment specifically for daily testing and another for testing what the live game would look like with addition of new content before the content is updated into the live environment.

WNF - Will not fix

2 What is Quality Control?

Quality control is an integral part of any game development or any software development company in general. Any game, software or application developed to be used by customers' needs to be validated before being released. The defects that users will potentially face are detected during the development phase as well as postproduction. Depending on the company, this department is commonly also regarded as quality assurance or specifically in game companies, as game testers.

According to Yeamin (2017), game "testers have to be obsessed with video games" (Hasanat 2017, 2) but the most important part of assuring the quality of the game is to be able to write the issue or bug in a clear and understandable manner.

The duties of quality control department are as follows:

- Confirming that the final product is consistent with the requirements and specifications before release (AltexSoft, 2016)
- Persistent and steady improvement and upkeep of the project. (Hutchison and Han 2007)
- Verifying game/software correspond to the rating and certification specified by the suppliers. (Business Dictionary 2007)
- Creating records for the issues discovered and assigning them to the correct developer for getting the issue resolved. (Job Hero 2018)

2.1 Skills required to be Quality Control

When one thinks of game testing, it is assumed that they need to have a degree related to the gaming industry or have many years of prior experience in the gaming industry. However, that is not true, according to Bay (2016), one does not even need a specific degree to become a game tester.

The key skills required to be a game tester are:

- Good communication - Communicating is a big part of being a game tester since discussing issues in the game with the developers or the specific developer for that particular issue is important to clarify what the expected outcome of the issue is. Sometimes the issue is NAD but to a tester from a consumer point of view it may actually be an issue, for example the UI of the menu could make sense to the person who developed it but to a consumer who will be seeing that UI for the first time may not know how to utilize it to navigate as designed. From a survey I conducted on May

2019 using SurveyMonkey a respondent who is a game artist has said “As developers we often become blind to our own errors.”

Not only do testers have to discover issues with the game from a content point of view i.e.: programming issues or art issues, they also must point out issues related to design. If some aspect of the game does not feel correct from a user point of view or seems unbalanced, then those issues also must be communicated to the designers before the design is released to the players. Hence discussing details like these is a crucial part of being a tester.

According to a Level Designer from a Finnish game company called Seriously, getting feedback on the fun aspect of the content they design is extremely useful, “sometimes more so than verifying bugs or difficulty balancing” (Survey 2019)

Good communication skills not only apply to verbal communication, but all written communication. A game tester must record issues that they discover and be as clear as possible on how to reproduce that issue so the developer can recreate that issue on their own device and understand what is wrong. Being comprehensible in writing is essential.

- Detail oriented - Having an eye for detail is a basic skill that a tester should have. Noticing small issues that even a consumer will probably not notice is an advantage that most testers would have. Even if the issue is not game breaking, it is always important to record those issues in case a similar issue arises in the future. According to Milla Karhu (Survey 2019), a QA tester from a Finnish game company called Remedy Entertainment, “This way, if a critical issue is encountered close to the end of development/close to a milestone, there is a higher chance of finding the root of the cause by finding similar issues”.

Noticing that an issue has been fixed by a developer is good. But what is even better is noticing what else has that fix changed within the game. Changing some code/modifying the art or UI within a game to fix an issue could also completely create another new issue. Knowing how to think ahead and expect certain issues that may arise is another major skill that a tester needs to have.

- Technically knowledgeable - Being a game tester means working around people who will be programming or people who are artists and creating content using different technical tools. Knowing what they are discussing is a skill a tester should have. Understanding why something is not working as it should be or why an issue is WNF is an ability that is desirable for a tester to have. Games usually have a development build which is the game used by the testers to test the game. This build will have debug

messages that may indicate what the issue is or help the developer understand what the origin of the cause is. Testers usually include these debug messages in the record of the issue they are making if needed. From a perspective of someone who has not worked with technology or has no experience with it may not understand what these messages mean. Including these messages will of course let the developer know what the issue is but if the tester has knowledge as to what steps they took that caused that error message to appear in the debug log then mentioning that in the record of the issue will make it easier and faster for the developer to pinpoint the root of the issue.

- Gaming knowledge - Finally, this is the most important skill of all. As mentioned, before you do not require a specific degree to be a tester in the gaming industry. However, playing games, knowing how a game works, knowing different platforms/devices that games can be played on, understanding gaming jargon are all usually essential to be a game tester. Most interviews for a game testing position will ask you what someone's favourite games are and on what platforms have they played. The games the person plays do not matter, it can be a game no one has heard of, what matters is that they know how to play on different platforms and know how to navigate game UI as well as understand the general functionality of a game. They need to grasp a new game and understand the general objectives quite easily. Furthermore, if someone does not enjoy playing games but must play the same game or a specific segment of the game several times, they may not enjoy doing their job as a game tester. A Designer/Producer (Survey 2019) has said "They should be able to recognize what is fun game play depending on the genre and remember that feeling as at the end of the development they usually have 2000-3000h playtime in the game and for a 20h game that's a lot of repetition."

Someone who has no interest in playing games will probably not be a good fit for a game tester, no matter how technically advanced, communicative or detail oriented they are.

2.2 Why is Testing Required?

Dani Jompero (Survey 2019), a development tester has said

"Testing is a tool that helps developers to deliver quality products by means of detecting issues and then reacting to them. It also allows developers to balance their workload if it is started early."

The main aim of this thesis is to determine the importance of having quality control in game development. To pinpoint how important it is, we need to first know why testing is needed in the development of a game.

Testing is the process of playing the game and locating issues. Testing is required when there is any playable or interactive content in the game to test. Testers can locate any issues so the problem can be fixed and not continue to occur until later phases of the game's development process. Testing is required in any situation when a product/software is to be created for consumers to use. Just like that, games need to be free from any issues before the end user can play it.

There is a lot of money at risk in the development of a game and for it to be successful. There are critics waiting to review new games for magazines and internet articles. Having negative reviews due to the presence of issues that could not be identified due to the lack of testing could prove to be expensive. The company could lose a lot of possible buyers due to the negative reviews related to bugs present in game.

From a consumer point of view, why would one wish to purchase a game if it contains a lot of issues? Of course, any software programmed will have issues, but the major issues are usually resolved before players get access to the game. The game can be improved, more issues can be resolved, and updated to the game even after the game is published. However, the initial impression a player will get when playing the game for the first time is very important and having major issues in a live game is unfeasible. Furthermore, when new content is released as DLCs, they are usually not free of charge. If the game plans on containing DLC but do not have any players interested in purchasing their game due to negative reviews due to the existence of bugs, then the initial plan to release these DLCs would also be unprofitable. According to Schultz, Bryant and Langdell (2005), if a game is going to be played by hundreds and thousands of players at the same time and they are expected to pay either monthly fees or for these DLCs then the game is expected to work as faultlessly as possible. (Schultz et al. 2005)

Not only testers may find the issues with the content in the game from the programming or art aspect, testers can also address issues with the gameplay itself. Testers usually have many hours of gameplay since they are playing the game several times a day or even just one section of the game repeatedly. (Simon 2015) As mentioned earlier, game testers are usually people who are interested in playing games. Testers can also inform the designers whether the plot of the game feel right or if the game is balanced in terms of character statistics. Information and suggestions regarding the gameplay will also help enhance the game before users can play with it. According to Milla Karhu (Survey 2019), the players are the top priority and the creative direction of the designer comes second to that. Which means that even if

there is a strict plan that the designer has thought out for the game it may or may not be correct in terms of playability.

According to a professional gamer from Finland, Leo (Survey 2019), "A buggy game decreases customer satisfaction and can potentially brand the game as "a buggy mess", by which he means the game will have a reputation of containing a lot of issues and possibly cause other potential players to not purchase the game.

Additionally, games that are being created for multiple platforms, for e.g.: PC, PlayStation, Xbox etc, the are expected to work on all these platforms as equally in functionality as possible despite the shortcomings of their hardware. There are different configurations and hardware that the game must be optimized for each platform. Console providers are the most important reason why the game should be bug free. Console providers/publishers are the people who publish the game on their respective platforms. They have their own set of testers who test the game for any issues before allowing the game to be released in their platform. Each console provider has different certification requirements which are a very strict list of things developers must conform to before they can release the game on those platforms. The game not only has to look, and work perfectly but also needs to follow the standards that are set by the console providers in terms of technicalities. An example is the game, *The Long Dark*, published by the company Hinterland Studio. The Creative Director, Raphael van Lierop (2017), outlined the process through which they had to go through to publish their game on Steam (a popular digital game distribution platform for PC), PlayStation 4 and Xbox One. For Xbox One they have mentioned that all their patches had to go through a five-day review process by the Xbox certification team. If any issues were found, then the entire process had to be started again and the users would not receive a patch on the Xbox platform. It is preferable to re-release patches at the same time on all platforms even if the patch has been approved by the certification teams on the other platforms, they need to wait till the patch has been approved on Xbox. However, in their case, the game on Xbox was one week behind the other platforms since they don't find out the issue with the patch till after the five days of certification process that is done by the Xbox certification team. Usually these patches are planned out in a way to keep in mind the different standards that are maintained by each platform. If there is going to be a delay, then the patch should be created early enough so they can all be released together in all platforms or as closely in timing as possible.

There are four main aspects to test:

Functionality/gameplay defect - This type of issue is when the user performs an action in game, but the action does not function the way it is supposed to. This type of issue will interfere with the gameplay and stop the user from continuing the game. However, some issues like this can also be misused and the player can use this sort of bug to progress in the game

faster than intended. For example: if the jump functionality is not functioning as it is supposed to then the player cannot pass a level. However, if the jump functionality is causing the player to jump twice the height than they are supposed to then the player can misuse it to clear obstacles more easily than it was designed to be. (Vasilis 2015)

Graphics Issue - This type of issue occurs when user experiences an issue with how the game looks. The issue could be related to the graphics of the objects within the game or the user interface of the game as well. Issues could be minor and not noticeable for example a misaligned object or major and very prominent for example an object clipping through another object. These sorts of issues could cause obstruction during gameplay for the player. If a player expects to drive a car through a certain object and the collision for that object is not set to the correct value, then the player could crash into that object and not complete the game. (CSAgents 2018)

Crashes and freezes - This kind of issue is when the game will shut down unexpectedly is known as a crash. When the game will cease to function and simply “freeze”, depriving the player the ability to control anything within the game is known as a freeze. Some functionalities within the game can cause the game to crash or freeze and these are some of the most major issues that developers tend to deal with first before any other issues. These situations cause the player to be unable to progress in the game. (CSAgents 2018)

Audio Issues - These issues are related to the sound effects and music in game. Some actions in the game should cause a certain sound effect to be triggered and when it does not it is considered to be a bug. According to Tommi Turunen (2019), Fortnite, a game created by Epic Games, has several audio issues. Fortnite is a shooting game where players win by being the only player alive after shooting every other player in the game. When there are enemies in the players vicinity, whether they are in a house, upstairs or downstairs, it is crucial for a player to hear their enemies' footsteps to defend or attack appropriately.

3 Game Development

Game development is the process of creating a video game. The process involves creating a concept for a game then developing it using programming, drawing, producing, recording and testing. When the word development is mentioned, it is usually thought that it refers to the content creators. However, testing is very much a part of the game development process.

The development part of the process refers to anything that aids in the creation of the game, whether it is designing the game, managing the process, programming the game, creating art, recording music and testing the game.

3.1 Different Roles in Game Development

Game development is the process of creating a video game. The process involves creating a concept for a game then developing it using programming, drawing, producing, recording and testing. When the word development is mentioned, it is usually thought that it refers to the content creators. However, testing is very much a part of the game development process.

The development part of the process refers to anything that aids in the creation of the game, whether it is designing the game, managing the process, programming the game, creating art, recording music and testing the game.

Before understanding the game development process and how quality control is involved in the process, it is important to understand the team and the roles involved in the process. Each game development company has different team sizes depending on the game. The roles may differ depending on the size of the company or the scope of the project. (H. Chandler and R. Chandler 2011)

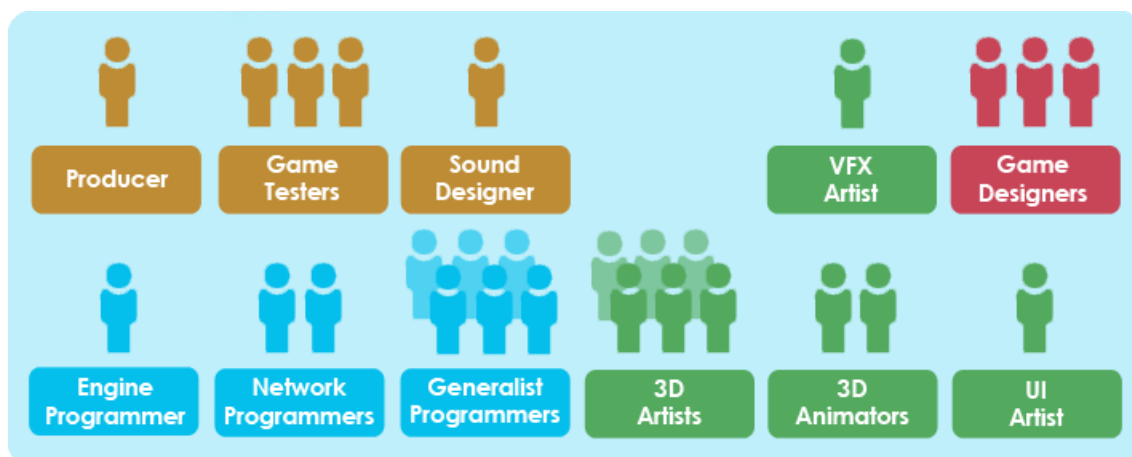


Figure 1 shows the general team a game development team would have (Rose 2014)

3.1.1 Producer

The role of the producer in the development of the game is to manage the entire process and to be the mediator between the development team and anyone outside the game development process. (H. Chandler and R. Chandler 2011) The producer usually is not involved in creating the content for the game, but they create schedules and keep the rest of the development team on track with the deadlines as well as dealing with marketing content, contract negotiation and communicating with the publishers.

There are different hierarchy within the producer role depending on the size of the company. These roles extend to Executive Producers and Associate Producers.



Figure 2 shows the people in the development team that are in contact with the producer. (H. Chandler and R. Chandler 2011)

John Hight an executive producer for the game Battleships described his tasks as “deciding which games to produce, which developers to work with and how much money should be given to the project” (Novak 2012a, 320). The producer needs to know the development team well and the technologies being used to create the game as well.

3.1.2 Designer

The game designer is responsible for the creation of the idea and concept of the game. Their main duty is to create an engaging gameplay with a captivating storyline. According to Chandler, “the designer is responsible for creating, prototyping, implementing, and balancing different areas of the game” (H. Chandler and R. Chandler 2011, 55). Depending on the game, there could be many different types of designers involved such as a Level Designer, UI designer, Narrative Designer and Systems Designer.

The main task that the game designer is accountable for that directly affects the job of quality control are the design documents they create regarding the concept and idea of the game. The testers can use these documents to make sure the content in the game is in accordance with how the designer wanted them to be. Designers provide documentation usually in the

form of a flow chart so the content creators as well as the designers know the flow of how the game must be played and what to expect in the next phase of the game.

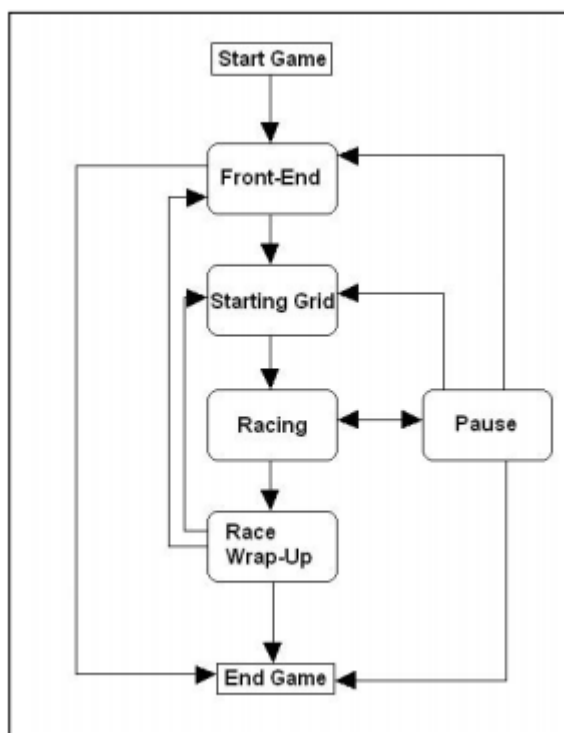


Figure 3 shows an example of flow from Street Racer. (Schultz et al. 2005)

This image shows what to expect when the user is playing the game and when the user can pause the game. Programmers can use this info to create the flow, so the user is restricted from being able to pause the game during the Race Wrap-Up but is able to pause the game in other phases of the flow. When a tester plays this segment of the game, they can verify that the user can only pause in those specific phases and not the unintended phases. Testers can also use this document to verify flow of the game and at which phases the user can restart the flow. Any deviation from this flow will result in the creation of an issue and be assigned to the programmer responsible for creating this workflow in game.

Along with documentation of the flow of the game, the screen layouts that should contain any information that the designer strategized must also be placed in a document for the UI artists/programmers to create and the testers to verify the output.

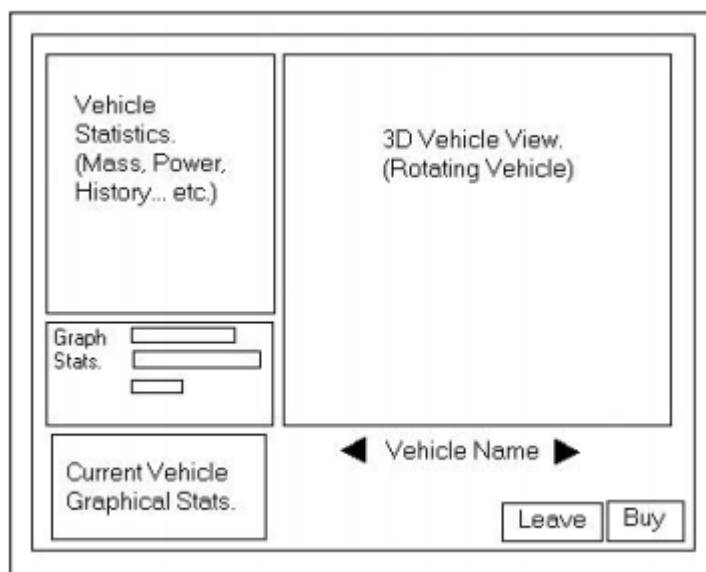


Figure 4 Shows an example from street racer. (Schultz et al. 2005)

In this image there is information regarding the screen layout for when a user wishes to purchase a vehicle. The placement of the images, text and statistics of the vehicle must be mentioned in this layout by the UI designer. The artists and the programmers can refer to this document to create this layout. The main purpose of this document is to let the content creators know exactly which information should be present.

When testers come across content in the game that may seem unusual, they can refer to these documents to confirm if they are accurate or not. If issues that are obviously inaccurate, for example an object in game has wrong texture, then an issue is created and assigned to the artist. When the artist resolves the issue, the tester can refer to the design document to verify whether the fix is corresponding to what the designer wanted.

3.1.3 Programmer

Programmer, or development engineer, is a general term for the different roles that are present in the development team. Programmers responsibility in general is to interpret the concept of the game created by the designers into code which in turn creates the playable part of the game. (Schultz et al. 2005)

Programmer may specialize in different fields in the game for example, graphics programmer, network programmer, interface programmer and tools programmer. (Novak 2012a).

The graphics programmer also known as the 3D graphics programmer is a combination of both an artist and a programmer. They use mathematics to create 3D graphics and environments that players can interact with. (Bethke 2003)

The network programmer is responsible for creating aspects of the game that will allow users to play the game with other users on the internet or via LAN (Local Area Network). Network programmers allow the game to be a multiplayer game by creating code that connects the game to the server and allows the game to communicate with the game on another user's platform. (Novak 2012a)

The interface programmer deals with creating the user interface of the game such as the HUD elements and navigational menus.

Tools programmer creates the tools that are used by the development team to integrate their content into code to comprise it in the game. The tools programmer works closely with the artists and programmers to understand what tools are needed by them.

It is important to understand the different roles of the programmers and which aspect of the game they are responsible for. This makes it easier for quality control to assign the issues to the appropriate person. However, depending on the company, the issues are usually assigned to the lead for each team and the lead will appropriately redirect the issue to the correct person since they have the knowledge of which programmer will have the best knowledge to resolve the issue or which programmer has enough time to deal with it.

3.1.4 Artists

Artists oversee creating all the graphical content for the game such as the characters, vehicles, buildings and level components. There are four general aspects that differ artists from each other, drawing, modelling, animation, and texture. (Novak 2012a)

2D and 3D artists are the creators of the objects in game. 2D artists create two-dimensional art assets that are used in the game. In a game that only uses two-dimensional art, a 2D artist has to create each image frame by frame if the character has to be moving. 2D artists are also needed in games that make use of 3D characters and objects. They can create the concept art for the game that 3D artists can use to make their models. They create character portraits or paint backgrounds as well as aid in the creation of marketing and website material. (Bethke 2003)

Animators are the artists responsible for adding motion to the characters and objects in game. Animations are composed of many sets of frames pieced together to create a smooth motion. Animated characters need to be moving in a realistic manner to look authentic. The animation needs to correspond to how the character look, for example their weight and

physiology. Not only the character has to move but their clothes and hair need to move along with their motion as well.

Texture artists create a skin that is used on 3D objects in the game. For example, different textures can be applied on a 3D cube to make it look like a wood box, a metallic box or even make it look like its covered in cloth. (Deetman 2013) Textures are not only used on static objects in game but also characters that are mobile which is why texture artists have to be precise in creating specific textures for different shapes in game as well as making sure that changing the size of the texture will not distort it. For example, if the character is modified.



Figure 5 shows the difference in the texture for the characters outfit when the texture options in settings are turned off. (Apex Legends 2019)

3.1.5 Audio

The audio team is in charge of implementing sounds into the game. Audio comes in three categories in game, sound effects, music and voice overs (Bethke 2003). Sound effects in game can be triggered when the player is interacting with various objects or even when navigating the UI. For example, if a user clicks certain buttons in game on the menu or pauses the game there will be a sound that is triggered to play. A different sound will be played if a door is opened in game or when its being closed. These audible experiences are minute but are

important to the game. Voice overs are audio team works closely with the designers and programmers to create the correct ambience needed for the game. (Schultz et al. 2005)

According to Winifred Philips (2019), a music composer for games such as Assassins Creed, she reviews design documents, views the parts of the game that are playable at that phase of development and makes decisions based on those to implement the category of music that would fit into those scenes. Usually the development team already has an idea about what sort of ambience they want to set for the game using music and she uses the vision they have to execute her plans.

Voice lines in the game are usually outsourced depending on the company and its budget. These audio clips are then used by the audio team to be manipulated depending on the scene of the game.

3.1.6 Quality Control

Testers are a crucial part of the development team. They test the content created by the rest of the development team to assure its quality. They test every aspect of the game from the UI, art, physics, collision, sound effects, camera angles, game progression, level playability, legal texts, localization, controller input, platform compatibility and much more.

There are two types of testing when it comes to content, positive testing and negative testing. Positive testing is when a tester creates a test plan to assess whether the game functions as it is intended to. For example, if a door opens and closes or if the UI navigates to the appropriate destination when user selects specific options. Negative testing is when a tester will purposefully attempt to “break” the game. Which means they will create unexpected situations in game to inspect whether the game will react appropriately or crash. For example, leaving the game turned on for an entire day and observe what will happen. Some game components do not handle the allocation of memory well and the game will freeze or cease. (Petronova 2015)

Testers will foremost locate bugs in the game and create reports for them. Secondary to that is to find issues with the playability of the game. When asked about testing the fun factor of the game, Dan Rossati (Survey 2019), a Creative Director from Ubisoft Redlynx, says that testers “are normally game enthusiasts and judge the game all the time, they are the best early indicators of a games longevity so I definitely take what is said in QA/QC on board and try to understand the feedback as best we can.”

However, according to Mădălina Vlădău (Survey 2019), a LiveOps tester, and Dani Jompero (Survey 2019), a Development Tester, the responsibility to measure the fun factor of a game is suitable to be carried out by the play testers. A playtest is the procedure carried out for a new game to locate bugs and design errors before releasing it. Playtests can either be open or

closed. In an open playtest anyone can play a limited amount of the game and provide feedback regarding it. Closed playtest is when only specific people are given access to the game. These playtests are done to test the game with several players on the live environment to obtain feedback regarding it. The players who are involved in the playtest of a game are usually not professional testers but the target audience the game is meant for.

4 Game Development Process and role of QC

There are 8 main phases that are involved in the production of a game, concept, pre-production, prototype, production, alpha, beta, gold and postproduction. (Novak) Depending on the project and the funds available, quality control may or may not be involved in most of these phases. This chapter will focus on explaining the phases and the role of quality control in each of these phases.

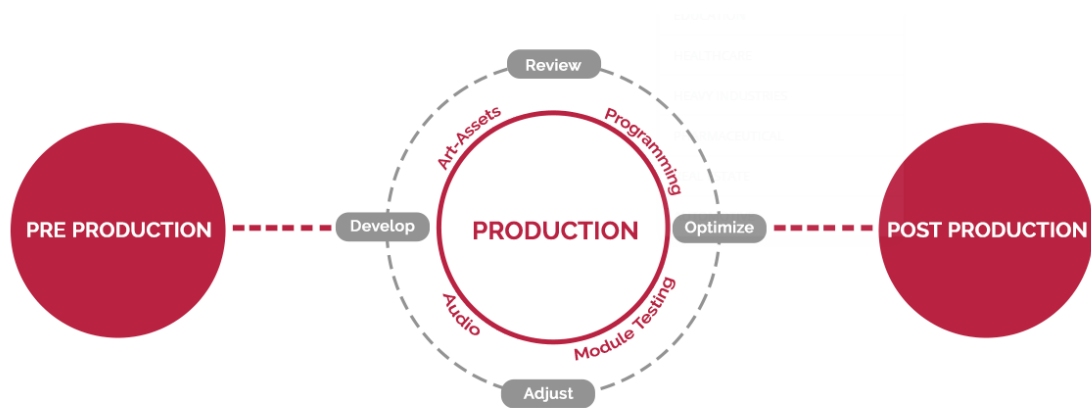


Figure 6 shows the general process done by game development company Juego Studios (2018)

4.1 Concept

At this phase of the game, there is no involvement of quality control. A designer, programmer, and an artist are typically the main people involved in this phase. (Novak) At this stage, the core idea of the game is conceived. The main target audience for the game is established and their thoughts and perception about why the game would be fun is investigated. (Laramee) Depending on the company, there is a pitching phase followed by the concept phase. In this phase the designer usually pitches the idea to publishers or the producer of the company using presentations that may involve concept art to aid in outlining the concept, genre, gameplay, storyline, target audience and possible platforms. The core intention of the pitching phase would be to demonstrate the why the game should be developed and why it would be successful. (Shultz)

4.2 Pre-Production

In the preproduction phase, the primary aim is to have a game design document by the end of this phase. The production path as well as the art style plan should also be conceived along with the project plan and finally a playable prototype. (Schultz) The budget estimate for the project is also planned at this phase to deduce if the project can be completed in the estimated time with the budget proposed. (Irish 2005)

Programmers, artists, producers as well as audio teams are the prime members involved in the pre-production phase. (Laramee) There is no requirement for quality control unless there is a playable prototype available.

4.3 Prototype

In this phase the “first playable” version of the game is produced. This is a tangible product that allows the people who weren’t involved in content creation, such as publishers, producers and investors, to understand the experience the game should provide. The main art style as well as the UI usability should be established as well as the core mechanics. (Laramee 2003)

If the prototype is not going to interest the publishers or investors within the first two minutes, then they are unlikely to move forward with the game. At this point the prototype also gives the testers a vision of what the game will be like and have a concrete game to test before the prototype is viewed by the non-development team. Verifying that the prototype does what the initial design document set out to prove is carried out by quality control. (Schultz et al. 2005) However depending on the size of the company and its funds, testers are sometimes not involved in this phase and testing is done by the developers themselves. Discussions on the requirements of quality control in a start-up company will be discussed in section 4.9.

4.4 Production

Production phase is the longest phase in the development of the game. Depending on the scale of the game, this could take from six months to two years. In this phase, the actual game is created with the aim of having a playable alpha game by the end.

At this stage, the emphasis moves from design aspect to technical and art aspects. However, this does not imply that the designer has nothing to do after the prototype phase since they must be constantly balancing the game and creating concrete future design documentations while the technical and art teams are producing the game infrastructure and visual features. (Laramee 2003)

Communication between teams is the most important aspect in this stage as well as having updated design documents, art plans and technical plans. The producer sets up a timeline with milestones the development team should follow. (Schultz et al. 2005)

In this phase of development having quality control and testing for the content being created is crucial. According to Tom Prinsen (Survey 2019), a Software Test Engineer from Eniram, testing should be done at all stages and automation should be emphasized as early as possible for each new functionality to detect regression tests as the project progress. The focus on the polished version of the game and UX can be shifted to later stages. At this point in the development phase, the important aspect to test and verify is the functionality of the game and validating the design documentation against the core mechanics and art concept of the game.

4.5 Alpha

The alpha stage of the development process is when the game is playable from start to finish. However, there is the possibility of the game containing bugs that are not game breaking and does not hamper with the flow of the game. There can be assets that are not finalized as well but the engine, UI and subsystems are finalized. (Schultz et al. 2005)

In this phase, the quality control department has a major part to play to make sure that each game component is functioning, and the creation of bug database and test plans are done. In some cases, play tests are performed in this phase allowed set number of players to have access to this unpolished version of the game to locate bugs that may be caused by a large amount of people playing the game. The development team should begin to resolve bugs that are in the database in this phase to begin polishing the game for the next phase. Tina Mäkelä, a Project Coordinator at Ubisoft Redlynx, says the crucial point in the development phase where testing is required is at the end of production phase.

There are some elements that must be fulfilled for the game to be considered at its alpha phase:

- Game play path is completed, where the game is playable from beginning to end
- Primary Language is localized into every game and no debug/placeholder text is present
- The game is in harmony with the preliminary documentation
- Compatible with the platforms the game is created for along with the hardware and software configurations

- Art and Audio present
- If applicable; multiplayer functionality is operative (Novak 2012b)

4.6 Beta

At the beta stage of the game the developers prepare to stop the addition of new content and only focus on polishing the game and resolving any bugs. The term “code freeze” occurs towards the end of this phase where all the work is done, and the game is being prepared to be converted to the master media usually in form of discs or final builds to be merged into game distribution software’s.

At this phase the role of quality control is to locate any issues related to the entire game that could be showstoppers or any issue that could cause the game to crash or freeze on specific platforms with different software and hardware configurations.

If the game is being developed for different registered platforms, then the testers need to ensure the game meets all the certification requirements since the publishers from those platforms will be conducting tests of their own to validate the final build being sent to them. The quality standards set by the respective platforms must be met to give the final builds a pass. (Schultz et al. 2005)

If the game is being marketed and sold around the world, then the game should have all the language localizations integrated into the game as well. There are localization testers, who are usually outsourced, who perform these tests. (Novak 2012b)

Associate Producer, Julius Fondem (Survey 2019), says the most important requirement for quality control is during the Beta to Gold Master stage of the development process. Several other responses from the survey also agree with this statement.

4.6.1 Open and Closed Beta testing

Within the beta phase of development, open and closed beta are also prioritized. Closed beta is when a set number of players are chosen to test the final version of the game and provide any crucial feedback. As well as gaining feedback from the players, the development team can also verify the functionality of the game when there are numerous numbers of players playing. They can also locate whether issues may arise from the different hardware and software the beta testers are playing the game on. (Schultz et al. 2005)

Open beta is when the game is released to the general public to gather feedback on the gameplay as well as market the game through the players who have signed up to play the game. Depending on the company, the open beta usually lasts a short time and the progress

gathered during the open beta is taken over to the officially released version of the game so players will have a head start on other players. (Skelton 2019)

If any major issues arise, the testers record the issues and assign any showstopping bugs to the development team and attempt to resolve it before releasing the game to the manufacturers.

4.7 Gold

At this phase the game is completed or almost completed. In recent times games are connected to the internet and new updates and patches can be added to the game even after the game is released to the players. The senior management and producers review the bug database and agree on allowing the game to be manufactured for release. Regardless of the issues left in game, the game is ready to be processed in the form of hardcopy CD's. Although in recent times the popularity of hardcopies has declined, and users prefer to purchase digital versions of the game which is available on all platforms.

However, the main priority is to ship a completed game so the players can enjoy the game without any issues with the flow of the game. The status of the game is inevitably depending on the management of the timeline created for the production phase up to the beta phase. If the timeline has enough buffer time available for developers to complete the game as well as resolve all the issues, then the game can be shipped as a complete game. James Strach (Novak 2012b) has said that the testing schedule set by the quality control department should not be viewed as buffer that can be easily squeezed into the last moment. Testing requires time and cannot be rushed, and enough time must be allocated to locate any issues that may have not been detected earlier. (Novak 2012b)

Julius Kantoluoto (Survey 2019), a build engineer from Ubisoft Redlynx has said "Games include an uncountable amount of moving parts with different kinds of software dependencies and relations so some of it are bound to break at some point. Testing the game reveals these weaknesses so they can be fixed before launching the game."

4.8 Postproduction

This phase is also known as "after release", where the game is continued to be polished and bugs are being resolved still. These fixes are released to the users in the form of patches which are free of charge. New issues are found due to the immense number of players playing the game and those issues are also dealt with and resolved. Updates are added to the game as well which are new contents that players can enjoy. These updates aid in the longevity of the game as well as player engagement and attracting new players as well. In many cases players don't purchase the game until months after the initial release date since they wait for the updated version of the game with the new content and the bug fixes.

In the case of multiplayer online games such as World of Warcraft, updates to provide new content is important since players are paying a monthly subscription and are expecting new experiences in the online world as well as complete fresh new quests and levels. (Schultz et al. 2005)

In a live operations game, there is new content being added in the form of seasonal events or challenges depending on the game. These weekly or monthly contents need to be tested as well before release on all the platforms with the different software and hardware configurations.

“In conclusion, thorough testing should be built into your schedule and prioritized throughout the course of development. Not to put too fine a point on this, but there is simply no such thing as too much testing.” (Wyman 2011, 176)

4.9 QC requirement in Start Ups

When a start-up game development company with limited funds is beginning the process to develop a game, do they need to allocate funds for hiring testers? Twenty-five people from related to the games industry have responded to this question. They were asked to measure the importance of hiring testers for a company with limited funds from a scale of 0-10. The results are shown in the chart (Survey 2019) below.

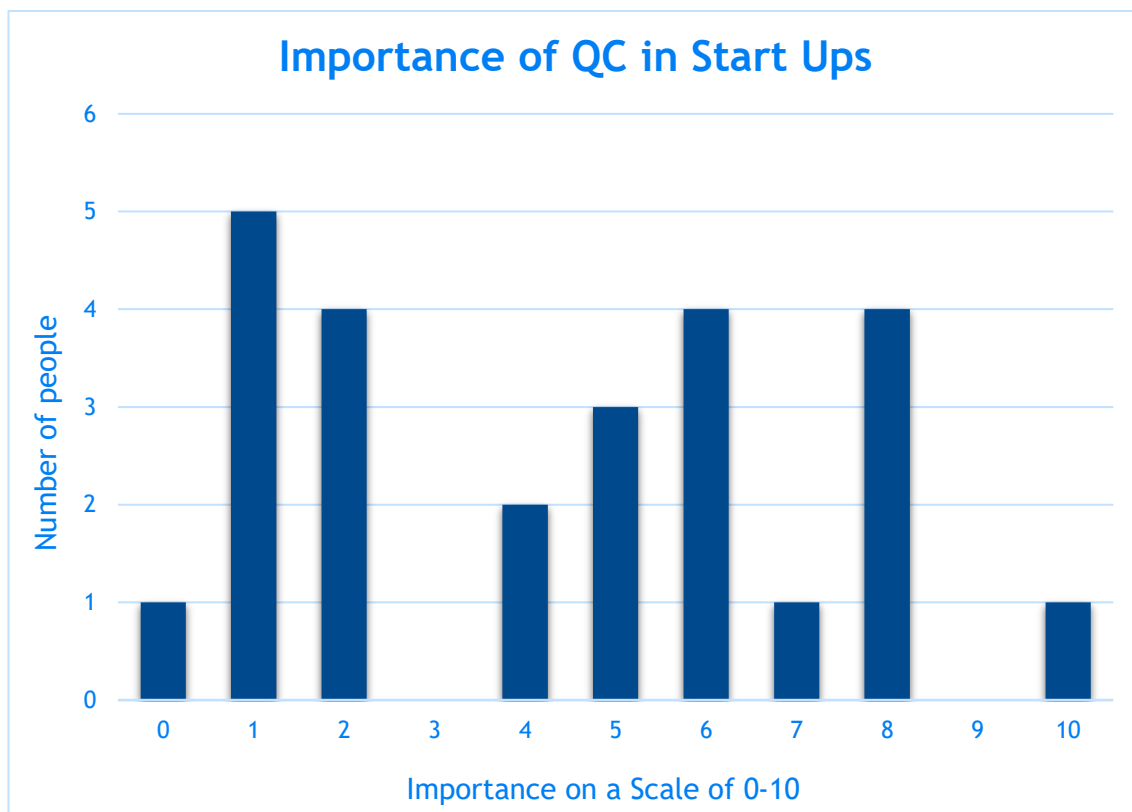


Figure 7 showing the survey responses regarding importance of QC in start-ups. (Survey 2019)

60% of the respondents selected the range of 4 or below which indicates that testers are not required for a startup with limited funds developing a game. A Junior Associate Producer from Ubisoft Redlynx says that programmers and artists are primarily required and should be invested on to create the base of the game. For these cases, programmers and artists need to do their own testing. The same ideology is agreed upon majority of the people who responded with a 4 or below in this scale of importance.

On the other hand, 40% of the respondents selected 5 or above on the scale of importance. Since it is a startup game development company in question who are creating their first game, Martin King, a Level Designer from Ubisoft Redlynx, has said that there is only one chance to make a first good impression, so it is important to have a tester to make sure the game offers a good first impression. Other respondents suggested if in-house testers cannot be hired by the startup then the testing should be outsourced to a dedicated testing studio.

Further investigation to the survey responses showed that majority of the respondents who are testers did not agree that a company with little funds should hire testers and majority of the respondents who were for hiring testers were content creators or producers. Most of the responses for why the start-ups should not hire testers was that developers can test their own content and have the entire company involved in testing the game. However, majority of the content creators who responded that the testing should be outsourced, or a single experienced tester must be present because the main priority of the content creators should be developing the game and not testing.

In conclusion, testing is a priority even if there is low funds and the company cannot invest in an inhouse testing team, testers can be outsourced, or the entire studio can be involved in testing the game as much as they can.

5 Quality Control in Depth

Similar to the other departments in the development of a game, the quality control department also has a hierarchy level set into place and different roles and responsibilities divided amongst each member.

5.1 QC Manager

The QC manager is the person from the testing team with whom the producer is in contact with majority of the time. They deal with creating schedules for the testing team with milestones on what content will be tested when and by whom. They also determine the size of the team depending on the scope of the project and the number of platforms that the game is published on. James Strach (Novak 2012b), QA Manager for Sony Computer Entertainment, says that his main responsibilities as manager is to create an environment that will enable his team to perform diligently and produce quality work. He also fosters communication amongst

the team members to feel comfortable approaching with issues as well as attempting to recognize personal growth and successes to keep high motivation and strong morale.

5.2 QC Lead

The lead tester is usually the most experienced tester in the team. They supervise the team and are involved in the daily testing processes and create test plans. They usually are the ones starting the process necessary to begin testing on the game and manage the data entry process, ensuring the testers are testing in the appropriate areas of the game and inputting accurate information. They also make sure the issues recorded have correct milestones assigned to them, so they don't get lost in the backlog. The lead tester also collaborates with the producer to determine whether a bug is important to resolve or not.

5.3 Compatibility Testers

These testers are primarily responsible to verify the game functions efficiently on different platforms. They not only test whether the game is running smoothly on each platform separately but also check whether the cross-platform functionality of the game works as well. They also need to test if the dedicated controllers for each platform function as well as test different controllers on PC. Since PC's can be customized unlike the consoles, they need to test the game with the minimum hardware specifications including video and sound cards. They are responsible for testing that the game meets the quality of standards set by the console providers as well.

5.4 Quality Assurance and Regression testers

These testers are involved in testing the daily builds of the game and aid with the development team to ensure every build they will be working on is playable and does not contain and showstopping bugs. They attempt to discover new bugs as well as verify the already resolved ones present in the database. They also are responsible for recording any new issues to the database discovered by the players or the development team.

5.5 Localization Testers

Localization testers focus on testing the linguistic aspect of the game and the translations. They also test the UI and UX, culturalization, internationalization and functional tests. They are native speaking linguists who review the game to ensure there are no errors in the translation of the language as well as if the content is culturally appropriate for the respectful country. These testers are outsourced either by freelancers or to a testing studio that. If the game is about to be published in multiple languages, every version should be tested by the appropriate native professional.

6 QC Processes

In this section the tools the quality control department utilizes to perform their duties will be discussed as well as the different varieties of bugs they have to deal with and how they are prioritized and recorded into the bug database



The figure 8 below depicts how a testing studio known as QAtestlabs (2017) perform their tests for a game

In the first step it is seen that the tester will analyse the system and outline the extent of what they will be testing. For example, if they will be testing an entire level or a single feature. The scope for testing a level is larger than testing a single feature such as testing character selection.

The second step would be to design a test plan where the team's goals are defined. The resources required for the test will be identified in this test plan as well as the methods that will be required to achieve it. The time required to test will also be mentioned to make sure the development team know when testing for the content is completed and when to expect results. (Schultz et al. 2005)

The third step is very important for tests that have a large scope. A test case is a document with test inputs, their execution methods and the expected output results. A set of conditions are verified by the tester to determine if the feature functions as the design document says it

should. (TestBytes 2017) This is a situation where having an updated game documentation is important for testers. According to Strach (Novak 2012b), materials provided by the developers and the game design document should be utilized to create test cases to make sure the game is functioning as designed.

User Interface	Check in Landscape/Portrait mode
	Check for animation, movement of character, graphics, Zoom In/Out (all gestures) etc
	There should not be any clipping (cutted background)
	Test whether one object overlaps with another
	Verify if loading indicator is displayed wherever required
	Character should not move out of the screen/specified area
	Test for enable and disable images/icons/buttons etc
	Check for screen title
	Check for message title, message description, label (should be appropriate)
	Check scrolling
	Font displayed (color, size etc)
	Check other objects too (ex -if its a car race- you need to look at road, people, other objects like buildings etc)

Figure 9 is an example of a test case to test the UI for a mobile game (TforTesting 2012)

Step four of the testing process is to execute the test case. As shown in the example above, the tester has to follow the test case and report the result for each situation. There are two varieties of testing, positive testing and negative testing. Positive testing is when the tester will input the correct value into the system, for example, if a door is expected to open in a level then the tester attempts to open it. This is how they identify the feature functions. Negative testing is when the tester will attempt to input incorrect values to test if the function works. If incorrect values are inserted. For example, if the door was only meant to open using the “open” button, but it also opens using the “close” or “jump” button then the tester will have to report that as an issue. Other methods of testing are discussed further in section 6.1.

The test cases are collected in one document, known as a checklist, to provide an overall status of the game and the result may be similar to figure 10 (Novak 2012b) shown below.

FPS MODE		Single Player		
Test	Results - 360	Results - PC	Bug #	Notes/Comments
Text	Fail	Fail	10007, 10206	
Audio	Fail	Fail	10007	
Graphics	Fail	Fail	10132, 10151	
Collision	Fail	Fail	10132, 10151	
Animations	Fail	Fail		
Main Weapon	Fail	Fail		
Secondary Weapon	Fail	Fail		
Vehicles	Fail	Fail	10186	
Particle Effects	Fail	Fail		
Interact - object	Fail	Fail		
Interact - AI	N/A	N/A		No interactable AI.
AI - animation	Fail	Fail		
AI - spawn	N/A	N/A		No AI enemies spawn.
AI - combat	Fail	Fail		
Triggers	Fail	Fail		
Resurrect	Fail	Fail		
Gameplay	Fail	Fail	10207	
Functionality	Fail	Fail		
Frame rate	Fail	Fail		
Cut Scene	Fail	Fail	10281	
Save	Pass	Pass		
Load	Fail	Fail		

Figure 10 shows an example of a test case. (Novak 2012b)

If the test case has failed in any aspect, then the bug is created and the issue number from the database is mentioned next to the test case to aid in providing more information where the issue has arisen.

Step five of the testing process is documenting the issues. As shown in the checklist figure, the issues that are documented need to be mentioned in the checklist document as well as the test case document. Documenting the issues is done by using a database tool, for example, Airbrake, JIRA and Bugzilla. These tools are shared among the entire team and all the bugs and their status can be seen as well.

information from the reporter then they can assign it back to the reporter with the status of NMI.

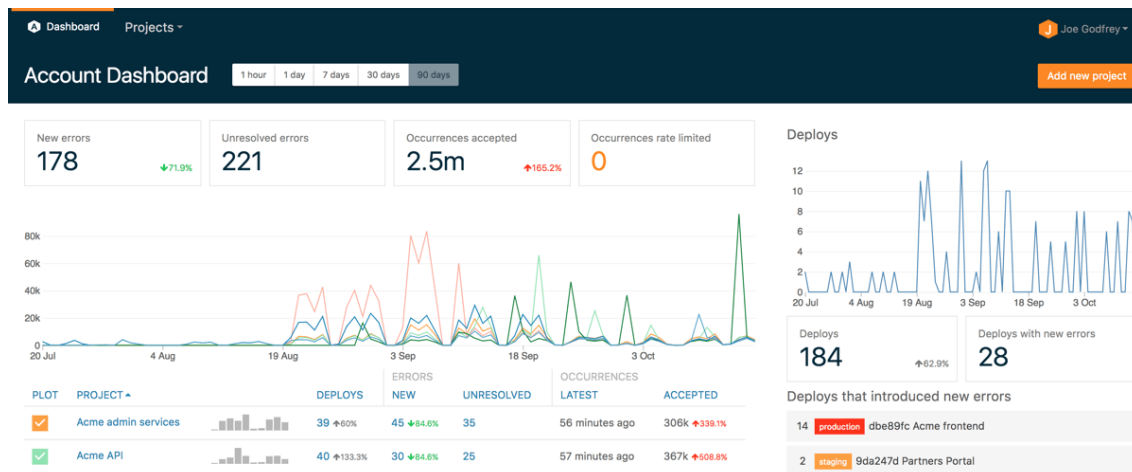


Figure 12 is an example from Airbrake showing the overall status of all the bugs in the database. (Airbrake 2011)

This dashboard can show how many unresolved issues are present as well as all the new issues recently input into the system. The dashboard can be filtered to show how many severe issues are present that require action or how many issues are still unresolved for the next milestone.

The final step of the testing process is to test for regression. When the issues are resolved by the developers, the tester will verify the fixes and will either close the issues if they are resolved or re-open the issue if not (Schultz et al. 2005). This method of testing also allows the tester to discover if the new changes added to the game causes any other part of the game to malfunction. (Smartbear 2013b)

6.1 Methods of Testing

Several methods of testing have been mentioned in this document but in this section, the different types of testing will be discussed in detail. The main types of testing that are generally conducted in majority of the game development companies are functionality, smoke, regression, ad-hoc, automated and progression testing. (Novak 2012b)

6.1.1 Automated Testing

As the name itself suggested, the testing done in this method is automated, which means the testing is done by means of using automation tools. Automated testing is best suited for larger projects where the same area of the game needs to be tested repeatedly. Using automation

tools saves time which allows testers to focus on other areas. (Smartbear 2013a) However the company will need to hire someone with knowledge in programming to create scripts within the game to support test automation. Test automation saves time, can emulate several players and give more consistent results but the tester must aid in this method as well and it's not completely automated. The results the automation tool provides needs to be processed by a tester and documentation still has to be written manually. Furthermore, the fun factor of the game cannot be judged by an automation tool so testing of that area in the game still has to be done manually. (Schultz et al. 2005)

Results on the chart (Survey 2019) below taken from the survey show that 36% of the respondents say that automated testing is not used in their workplace while the same percentage of respondents say that both manual and automated testing are used. Only 8% of the respondents say that only automated testing is used in their company while the rest of the 20% preferred not to answer.

USE OF AUTOMATED TESTING

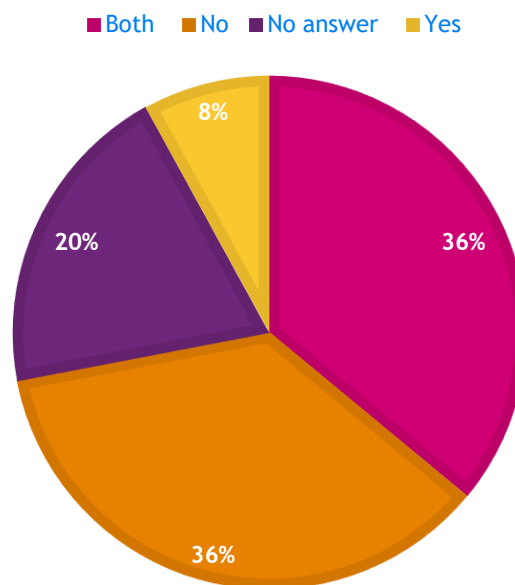


Figure 13 shows the usage of automated testing in game companies taken from the survey. (Survey 2019)

6.1.2 Functionality Testing

Functionality testing is when the tester will test the game to identify any issues related to the game play. Issues such that could cause crashes or freeze are uncovered. Any bug that causes the progression of the game are searched for too. Similarly testing all areas of the game for graphical errors, audio issues, collision, physics, and triggers are tested. Texts within the game are also verified to discovered spelling errors and inconsistencies. (Thipse 2014)

In this process the only focus is to test the overall functionality of the game and not the design or fun factor. (Novak 2012b)

6.1.3 Progression Testing

Progression testing is when the tester plays the entire game or specific levels to complete it. The main aim of this test is to finish the game from start to finish without any situation causing the tester to be unable to complete it. The game crashing or freezing is not one of the situations that is considered but the game design is. For example, if the tester has to complete the level by opening a door but the door does not open, then the tester cannot pass that segment of the game causing the inability to progress. These are the sorts of issues the tester is keeping a look out for when performing progression testing. (Novak 2012b)

6.1.4 Smoke Testing

When a new build is generated with all the fixes and new content, the tester needs to verify the build is functioning and all the basic functions of the game are working. The general cases in a smoke test is to check the game will load with an existing user account as well as a new user account. The tutorial of the game must function and the first few levels must be progressed to check for any issues with unlocking the next level. If the build is functioning without any crashes or freezes, then the smoke test report is sent to the team to verify that the build is ready to be used. Any issues that do not interfere with game play will be mentioned in the test as well to make sure they get fixed on the next build. The tester can continue use this verified build for regression testing. (Schultz et al. 2005)

6.1.5 Regression Testing

As mentioned earlier, regression testing is the process of verifying the bugs resolved by the developers. Testers will load the build mentioned by the developer where the fix has been integrated into and test if the issue has been fixed. The tester will also test if any other issue has arisen due to the fix from the developers. (Smartbear 2013b)

6.1.6 Ad-hoc Testing

Ad-hoc testing, also known as free testing, is the form of testing when the tester plays the game without the use of test cases. They play game however they please whilst keeping in mind to notice any bugs. This form of testing helps the tester find issues that they wouldn't be focusing on if they were following a test case. Issues that may arise with players in the live environment are usually discovered this way.

6.2 Tools used by QC

The daily tasks of someone from the quality control department does not only consist of the game and utilizing the bug reporting database. Testers must have extensive knowledge on

how to use several tools depending on the different types of platforms the game is published on as well as possible internal tools specifically created for the game.

There are tools that help with the reporting of an issue and there are tools that the tester must know how to use to install builds into the development kits.

6.2.5 Bug recording tools

When a bug is reported, the reporter has to provide visual aid for the developer to indicate where the issue is occurring. The most popular gaming devices and platforms will be mentioned in this section.

The easiest method to record an issue occurring on screen is to use a screen capturing software. On Apple mobile or tablet devices, the operating system comes with an option to record the screen. On Android mobile or tablet devices, Google Play Games has screen capture functionality.

When it comes to PC, software can be downloaded such as OBS Studio. (Ellis and Wilson 2019) This software allows the user to capture a specific window and save the recording onto their computer. The video can be trimmed to focus on a specific point of the video by using editing software such as Movavi.

On the PS4, the controller comes with an inbuilt function for screen capture. Similarly, the Xbox One and Nintendo Switch also contain screen capture functionality. (Miranda 2019) The videos from both the devices can be uploaded onto the computer to be edited and uploaded to the bug reports. (Marinconz 2014)

These are methods of recording the video capture for the game and similarly, screenshots for issues that don't need to have a video attachment can also be done on these devices. All the devices mentioned above contain inbuilt screenshot functionalities as well which makes it easy for testers to record issues visually.

6.2.2 Platform Specific Tools

The testing team does not utilize retail consoles for the purpose of testing development builds, but they use game development kits (GDK). These kits are only available to developers who are in agreement with platform manufacturer.

Each of these development kits need to be used with its respective software to install development builds created by the developers. The development build is the version of the game only accessible by the developers to develop with and to test on.

The following tools are needed to be used by the testers to test the builds on the development kits:

- PS4 Neighbourhood for PlayStation 4 is a graphical user interface through which developers can use their PS4 development kit with. (PSDevkWiki 2016)
- Nintendo Dev Interface for Nintendo consoles, helps developers set up their development environment on the Nintendo development kit. (Nintendo 2016)
- The Xbox Device Portal for the Xbox is a device management tool that allows a developer to add their game to the development kit and utilize it. (Satran, Cowley, Estabrook, Jacobs and Koren 2017)
- Xcode is an integrated development environment that can be used by testers for Apple devices. A testing engine is built into this to aid in testing UI and other performance tests. (Apple 2018)

These are the general tools that a games development company may use, however depending on the company there are many more tools that are used by the testing team to record performance statistics such as FPS measurements, console viewers to analyse log files, console management tools and console debugging tools.

6.3 Types of Issues

As discussed briefly earlier, there are different types of bugs that testers must look out for. There are eight main categories of bugs: visual, audio, design, physics, stability, networking, performance, monetization and compatibility.

6.3.1 Visual bugs

These are issues caused by graphical errors in the game. Issues can be prominent or not visible at all.

- Clipping is when two objects in the game move through one another. An example is figure 14 (Glacier Games 2013) below where the sword is clipping through the character.



Figure 14 shows objects clipping in game (Glacier Games 2013)

- Screen tearing is a common visual issue that occurs when the video feed is not sync with the displays refresh rate and noticed more often when the video is moving horizontally, for example if the camera is panning from left to right. Basically, the GPU (graphical processing unit) begins to draw the next frame before the current frame is still displayed. Figure 15 (Burke 2015) below is a noticeable example of screen tearing in a game.

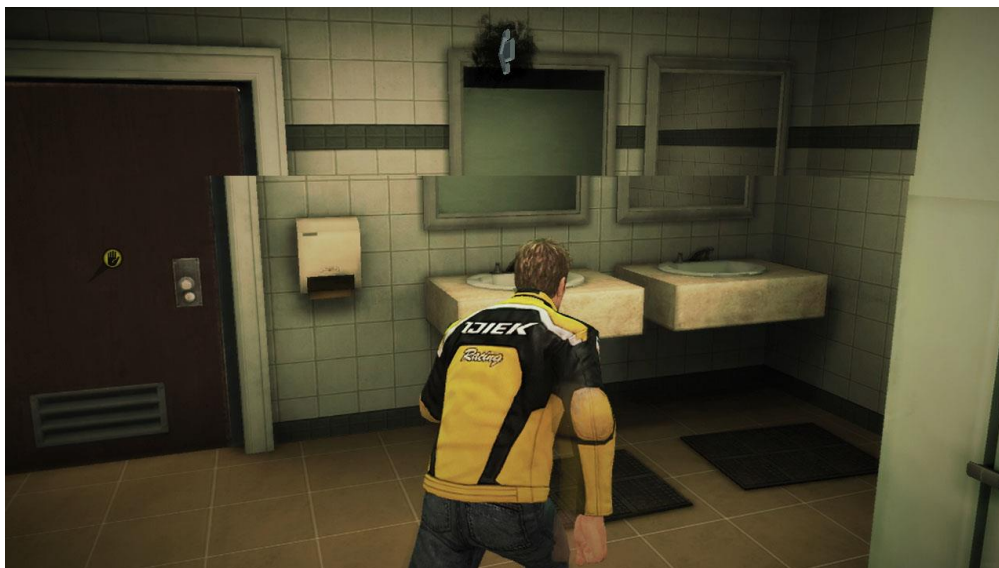


Figure 15 showing example of screen tearing. (Burke 2015)

- Missing textures is when the texture of a 3D object in game is shown as a solid colour, usually white and sometimes appears when the game engine realizes its missing. These issues are not severe if they are occurring on objects that aren't in the primary view of the player. However, if an object the player has to interact with as part of

the progress is missing textures, then that issue is considered severe and is resolved as soon as possible. (Novak 2012b)

6.3.2 Audio Bugs

Audio bugs are issues caused by missing sounds, audio drops, volume levels and skipping.

- Audio drop is when the voice line, music or sound effects miss and entire segment causing the player to miss dialogs or important sound effects in game.
- Bugs with volume levels is when the sounds playing at a specific moment suddenly sounds louder or quieter. Sometimes dialogs can sound fainter than the music and players can have issues attempting to understand what is happening
- Skipping is when the sound effect has an obvious distortion when being played.
- Missing sounds as the name suggests is when the music or sound effect is absent. Sometimes issues related to missing sounds can be easily identifiable for example if an entire level does not contain music. While in other cases it can be quite difficult to detect this issue for example the sound effect of the menu closing when the player presses a button to close the menu.

6.3.3 Level Design

Level design is the process of designing the environment the player will progress through. It could be one open world map or a single level. Level designers create these environments keeping in the mind the difficulty level for them and how easy it should be for the player to progress through. Nevertheless, there can be issues present with the design itself such as areas where the player may get stuck due to the miscalculated geometry, for example: a character getting stuck between two boxes in an attempt to conceal oneself. The geometry of those boxes may look linear; however, they aren't, and the character may not find the way out of that spot.

Another level design issue could be a "hole in the map". This issue is when there is an actual hole on the ground in the level. It could be visible or could be a regular looking patch of ground with no collision where the player can fall through. These holes can be exploited by players if they use these holes to conceal themselves from other players or they can visual distorting if the player simply falls through and the entire level disappears.

Invisible walls are another sort of issue that could occur where the character could be moving through an empty space but collides into an invisible wall. This can be a severe issue if the progression of the level requires the character to go through that specific area and the player cannot. (Novak 2012b)

6.3.4 Issues with physics

Physics in a game is the simulation of real-world physics so the movements seem realistic. A program code is used to simulate Newtonian physics known as the physics engine. Collision detection is also utilized to determine the course of action after two objects collide. When objects or the environment within the game do not follow the physics and collision rules then it is considered a bug. For example, a character placing a glass onto the table but instead the glass falls through or ball being kicked towards the goal but instead it simply moves vertically instead of horizontally. (Reilly 2012)

6.3.5 Stability issues

Stability issues are regarding bugs that occur when the game crashes, freezes or fails to load. As described earlier, crashing is when the game closes unexpectedly or the display turns blank and the user has to manually close the game through their computers task manager.

Freezing is when the game will stop responding and the display will be stuck. The user cannot close the game since it will not respond to any input functions. The user must close the game manually in this case as well.

When a game loads several megabytes in the console's memory is loaded as well very fast. At this instance, the game code can be halted from loading causing the user to be caught at the loading screen for an infinite amount of time. The user will have to restart the game manually for this case as well.

6.3.6 Network bugs

Network bugs are related to server client connectivity and bandwidth issues. Issues can result in failed connection, dropped connections, and lag.

Failed connection is when the user is attempting to connect to the server to possibly join a game and the game refuses to connect due to server or bandwidth issues.

Dropped connection is similar to failed connection however in this case the user manages to connect to the server but gets disconnected from it at some point. This is considered a worse situation than a failed connection since the user may lose progress they have gained before the connection was lost

Lag is when there is high latency in game. Latency is the time it takes for the user's device to communicate with the server and back to the user's device. High latency means the amount of time it is taking for the communication with the server is a lot which is not desirable. This causes the game to suddenly pause and then resume after few seconds and the player loses control of the game at that moment. The actions of the player will appear to have happened few moments later than they anticipated for example firing a gun at a target, but the target has moved away from that spot. Due to high latency, the user's character in game will shoot that spot after a delay causing the player to miss the target.

6.3.7 Monetization Issues

Purchases in games and IAPs are becoming more popular in the recent times. This is when the user can purchase items or soft currency (currency in game usually gained by progress) by using real currency. Users can pay online to accept their in-game items or currency. Sometimes items can be purchased to help the user progress faster than other players. However, there can be issues with monetization where the user cannot complete the process of purchasing the item. Another issue could be the delay in receiving the purchased item which would cause the player to assume they have paid for nothing. Sometimes the players could have a failed purchase scenario and lose money and not receive items due to a bug. In these cases, the players usually contact customer support to resolve any missing items or refunds, since these issues are dealing with expenditure from the players part. (CSAgents 2018)

6.4 Bug Prioritization

When issues are recorded into the database, they are assigned a priority level to determine the severity of the issue. The severity of the issues and their names may differ from company to company, but the general definitions are same. There are two aspects to the prioritization that is given to these issues. The impact the issue will have on the game and the priority as to which it must be resolved. As shown in figure 15 (Software Testing Help 2019) below, the tester usually defines the severity of the issue while the producer will define the priority depending on the milestone the issue needs to be resolved by. In quality control, the testers usually have knowledge of the milestones the issues need to be resolved by and define both the aspects themselves.

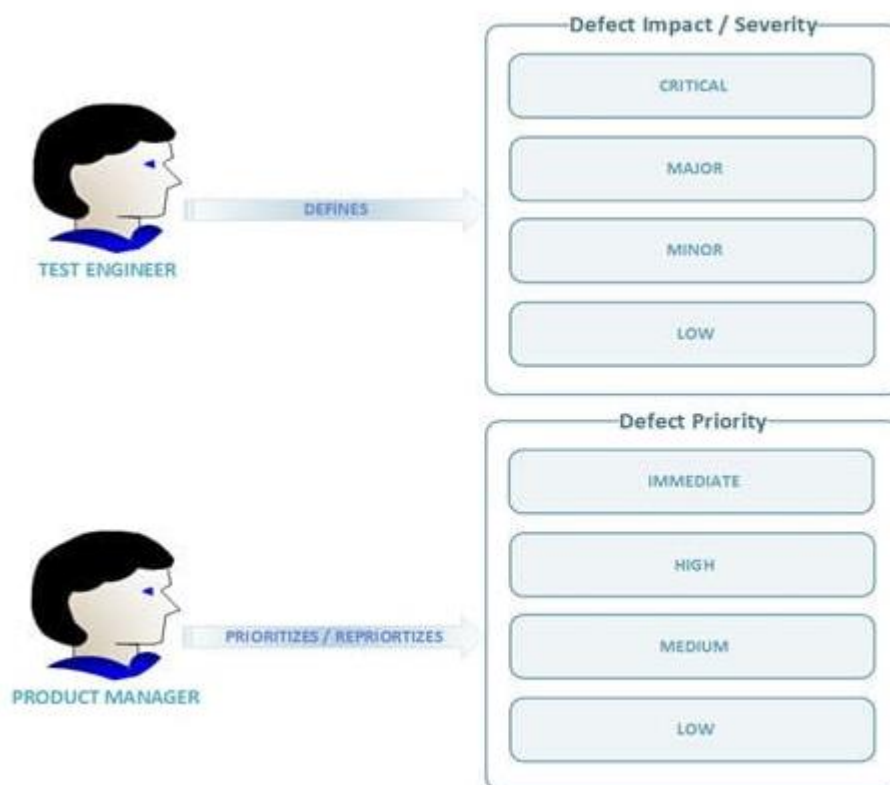


Figure 16 shows the bug severity and priority levels (Software Testing Help 2019)

Critical - The most urgent issues can be classified as severe issues where the issue is game breaking, for example: a bug that causes the game to crash. (Schultz et al. 2005) Critical issues such as these are resolved as soon as possible and usually never present in the gold master version of the game.

Major - Issues that have been labelled as high priority bugs are issues that affect the gameplay in functionality and performance. Issues could occur in FPS drops or the user not receiving their reward for completing a level. These issues sometimes make their way into the final version of the game only to be discovered by players. Sometimes these issues cannot be reproduced in the development environment and the presence of many players cause these issues to arise. (Schultz et al. 2005)

Minor - These issues are the most common issues that occur in the game but are not hindering the player from progressing in the game. These issues are the first to get resolved for the next patch after all the severe and high priority bugs are resolved. Sometimes issues with major severity are reduced to minor status if their priority is reduced.

Low - These issues have low probability of occurring but are still present in game. For example, these issues may occur with a specific combination of hardware by a user or a spelling error discovered in a pop up. (Novak 2012b)

6.5 Documentation

One of the main skills a tester is required to be knowledgeable about is the usage of bug tracking software. This software allows the tester to report and track the status of the issues in the project. Every project will have a new defect database and the testers must know how to create and edit them. (Petronova 2015)

Before writing a bug report, the tester must search the database using key words related to the issue to investigate whether the same issue has been reported by another tester before. If such an issue is found and the issue has the status of being resolved, the tester must reopen the issue and place a comment containing new information regarding the issue. (Sloper 2013)

When an issue has been discovered that is not already in the database, the tester has to create a report to assign to the developer who is responsible for fixing the issue. A bug report contains the follow main elements:

Summary or Title - The summary contains a short description regarding the issue, the area of the game the issue was discovered in and the type of issue it is. This is so the developers can get brief information about the issue and whether they will continue with the issue or assign it to someone else.

Description - The description of the issue will contain more information regarding the bug. Any information regarding the hardware, software, build number and sandbox will be mentioned in this field.

Expected outcome - The tester will refer to the design documentation and report what the expected result should be in place of the bug.

Reproduction steps - This is the most important segment of the report which contains step by step what the tester has done for the bug to occur. The steps must be written as clearly as possible, so the developer knows how to recreate the scenario on their workstation.

Severity - The severity of the issue and well as the prioritization must be included in this field.

Attachments - Any attachments that will aid the developer understand where the issue is should be included in the form of images, videos or error logs. For example, If the game shuts down when the user selects a specific menu then it is self-explanatory, and the developer can

reproduce this issue by following the steps, then no attachments are required besides error logs. However, if the issue is that the character in the game will walk through a specific wall in a specific area of the game, then the tester must include a video or image of the scenario so the developer can visualize which area it is precisely occurring in.

7 Research Methods

Two types of research methods were approached in this thesis, quantitative and qualitative

Qualitative Research - The main method used in this thesis is case study method. According to Yin (2003) case study design should be considered when:

- The focus of the study is to answer “how” and “why” questions, in this case the thesis is outline how the quality control department carry out their tasks and why they report issues and to whom.
- You cannot manipulate the behaviour of those involved in the study, none of the information used in this thesis is manipulated. All the information is taken from existing documents, books and survey responses from individuals involved in the gaming industry. They were given the option to not reveal their names and their companies so they can provide honest responses to the questions asked.
- You want to cover contextual conditions because you believe they are relevant to the phenomenon under study, in this case the study is about resolving the misconceptions regarding the tasks and responsibilities of the quality control department. The knowledge, skills and their duties are outlined to show they are equal in terms of other members of the development team.

Secondary qualitative methods were also used in this thesis where the available data on the internet was researched. Articles written about the quality control department of the game development, books obtained from the public library, commercial journalistic articles and three main book references were used.

Quantitative Research - This method of research is used to quantify the theory by the means of using data collection methods and analysing existing data available. The data is used to quantify attitudes and opinions towards the topic. This data is structured used measurable data to gather information. (DeFranzo 2011)

A survey was taken of 25 people involved in the gaming industry to gather information from different fields of the development team on their opinions on the quality control division. Two graphical charts were created to visualize the responses of the respondents and several

responses were quoted in this thesis as well were relevant. Both open ended and close ended questions were asked to gain measurable data as well as opinions of the respondents

8 Conclusion

An article titled “Want to get paid to play games? Become a game tester!” is quite exciting to see for a person who plays games. Reading this line “Game testing from home is now a new trend to earn money” from an inexperienced point of view might sound like testing games is simply the act of stay at home and playing games while getting paid (Malini 2019)

The aim of this thesis was to outline the importance of quality control in game development and outline the tasks and skills a tester must have to be a part of the development team. Testing is viewed as an easy job that any player would enjoy doing. The term “testers get paid to play games” has been used over the years by anyone playing games or even developing them.

This thesis is written to describe the quality control process and to understand its importance in the role of the entire development process. The types of issues faced and the methods to discover them are also analysed. As well as the tools required to find issues in the game and the extensive knowledge the tester must possess to utilize them. The professional aspect of being in quality control requires more problem-solving skills than trying to get top score in a game. According to Jason Schreier (2017), testers are underappreciated and only recognized when an issue is discovered by a player in game Their tasks might look repetitive and monotonous but in reality, it is stressful and tedious and requires a specific “attitude and outlook” to be a good tester. A game testing profession is usually seen as a gateway to the gaming industry to continue onto other viable jobs instead of considering that quality control itself is a viable career path. A level designer (Survey 2019) from a Finnish game company, Seriously, has said not to treat game testing as a starting point to a more “glamorous” job such as coding or design. Testing should be taken seriously as it is its own career.

In conclusion, the main purpose was to resolve misinformation people in the gaming industry as well as people outside of it have regarding the duties of a tester and the large impact they have on the final game.

References

Printed sources

Bethke, Erik. 2003. Game Development and Production. Plano, Texas

Chandler, Heather Maxwell and Chandler, Raphael. 2011. Fundamentals of Game Development. Sudbury, MA.

Laramee, Francois Dominic. 2003. Secrets of the Game Business.

Novak, Jeannie. 2012a. Game Development Essentials an Introduction 3rd Edition. Clifton Park, NY

Novak, Jeannie. 2012b. Game Development Essentials Game QA and Testing. Clifton Park, NY

Schultz, Charles P., Bryant, Robert, Langdell, Tim. 2005. Game Testing All in One. Boston, MA.

Wyman, Michael Thornton. 2011. Making Great Games Burlington, MA.

Yin, Robert K. (2003). Case study research: Design and methods (3rd ed.). Thousand Oaks, CA

Electronic sources

AltexSoft. 2016. Quality Assurance, Quality Control and Testing – the Basics of Software Quality Management. Accessed 17 May 2019. <https://www.altexsoft.com/whitepapers/quality-assurance-quality-control-and-testing-the-basics-of-software-quality-management>

Apple. 2018.Xcode. Accessed on 2 June 2019. <https://developer.apple.com/xcode/>

Business Dictionary. 2007. Quality Assurance (QA). Accessed 17 May 2019. <http://www.businessdictionary.com/definition/quality-assurance-QA.html>

CSAgents. 2018. 7 Most Common Video Game Glitches and Bugs, and How to Improve Player Experience. Accessed on 26 May 2019. <https://cs-agents.com/blog/common-video-game-glitches-bugs/>

Deetman, Koen. 2013. Different Types of Game Artists. Accessed on 29 May 2019. https://www.gamasutra.com/blogs/KoenDeetman/20130730/197246/Different_Types_of_Game_Artists.php

DeFranzo, Susan E. 2011. What's the difference between qualitative and quantitative research? Accessed on 3 June 2019. <https://www.snapsurveys.com/blog/qualitative-vs-quantitative-research/>

Ellis, Catherine, Wilson, Mark Wycislik. 2019.The best free screen recorder 2019: record or stream all the action. Accessed 2 June 2019. <https://www.techradar.com/news/the-best-free-screen-recorder>

Hasanat, Yeamin. 2017. Game Testing Introduction. Accessed 15 May 2019.

<https://www.utest.com/articles/game-testing-introduction>

Hutchison, Allen and Han, Jay. 2007. The difference between QA, QC, and Test Engineering.

Accessed 17 May 2019. <https://testing.googleblog.com/2007/03/difference-between-qa-qc-and-test.html>

Job Hero. 2018. QA Tester Job Description. Accessed 17 May 2019. <https://www.job-hero.com/qa-tester-job-description/>

Lierop, Raphael van. 2017. Update on Build Status for Xbox, PS4 & Patching Process. Accessed

25 May 2019. <http://www.hinterlandforums.com/forums/topic/16657-update-on-build-status-for-xbox-ps4-patching-process/>

Malini. 2019. Want to Get Paid to Play Games? Become A Game Tester! Accessed on 3 June

2019. <https://www.softwaretestinghelp.com/get-paid-to-play-games-become-game-tester/>

Marinconz, Steve. 2014. How to Capture Video Game Footage on PS4, Xbox One, PC and Mac.

Accessed on 2 June 2019. <https://kotaku.com/how-to-capture-game-video-on-ps4-xbox-one-pc-and-mac-1659764485>

Miranda, Felicia. 2019. How to Capture Gameplay on the Nintendo Switch. Accessed on 2 June

2019. <https://allgamers.com/article/2402/how-to-capture-gameplay-on-the-nintendo-switch>

Nintendo. 2016. Tools/Middleware. Accessed 2 June 2019. [https://developer.nin-](https://developer.nintendo.com/tools)

[tendo.com/tools](https://developer.nintendo.com/tools)

Petronova, Ilieana. 2015. Game Testing Strategy. Accessed May 30 2019

<https://www.linkedin.com/pulse/game-testing-strategy-ilieana-petronova/>

Philips, Winifred. 2019. Composer Winifred Phillips answers Reddit's questions in viral Ask-Me-

Anything about video game music. Accessed on 29 May 2019. [\[tra.com/blogs/WinifredPhillips/20190528/343408/Composer_Winifred_Phillips_answers_Red-dits_questions_in_viral_AskMeAnything_about_video_game_music.php\]\(https://www.gamasutra.com/blogs/WinifredPhillips/20190528/343408/Composer_Winifred_Phillips_answers_Red-dits_questions_in_viral_AskMeAnything_about_video_game_music.php\)](https://www.gamasu-</p></div><div data-bbox=)

PSDevWiki. 2016. Neighborhood for PlayStation 4. Accessed on 2 June 2019.

https://www.psdevwiki.com/ps4/Neighborhood_for_PlayStation_4

Reilly, Luke. 2012. The Most Impressive Physics Engine You've Never Seen. Accessed on 2 June

2019. <https://www.ign.com/articles/2012/10/01/the-most-impressive-physics-engine-youve-never-seen>

Satran, Michael. Cowley, Eliot. EstaBrook, Norm. Jacobs, Mike. Koren, Alexander. 2017. Intro-

duction to Xbox One tools. Accessed 2 June 2019. [https://docs.microsoft.com/en-us/win-](https://docs.microsoft.com/en-us/windows/uwp/xbox-apps/introduction-to-xbox-tools)

[dows/uwp/xbox-apps/introduction-to-xbox-tools](https://docs.microsoft.com/en-us/windows/uwp/xbox-apps/introduction-to-xbox-tools)

Schreier, Jason. 2017. Quality Assured: What It's Really Like to Test Games for A Living. Ac-

cessed on 2 May 2019. [https://kotaku.com/quality-assured-what-it-s-really-like-to-play-](https://kotaku.com/quality-assured-what-it-s-really-like-to-play-games-fo-1720053842)

[games-fo-1720053842](https://kotaku.com/quality-assured-what-it-s-really-like-to-play-games-fo-1720053842)

Simon. 2015. Game Testing: Why It's Important to Invest in a Game Tester. Accessed 21 May

2019. <https://leantesting.com/importance-in-finding-game-tester/>

Skelton, Jamie. 2019. Closed Beta? Open Beta? Early Access? What It All Means. Accessed 30 May 2019. <https://mmohuts.com/news/what-does-closed-beta-mean/>

Sloper, Tom. 2013. How to Write a Game Bug Report. Accessed on June 3. <http://www.sloperama.com/advice/faq75.htm>

SmartBear. 2013a. What is Automated Testing? Accessed 2 June 2019. <https://smartbear.com/learn/automated-testing/what-is-automated-testing/>

SmartBear. 2013b. What is Regression Testing? Accessed 2 June 2019. <https://smartbear.com/learn/automated-testing/what-is-regression-testing/>

Survey.2019. Responses of Survey - Importance of Quality Control in Games Development. Created 6 May 2019. https://drive.google.com/open?id=1_12dLichcdJvpefPlfaKa_xEU-f5RYNF

Testbytes. 2017. How to Create Test Cases in Android Application. Accessed 30 May 2019. <https://www.testbytes.net/blog/create-test-cases-android-application/>

Thang, Jimmy.2012. The Tough Life of a Games Tester. Accessed on 2 May 2019 <https://www.ign.com/articles/2012/03/29/the-tough-life-of-a-games-tester>

Thipse, Laxmikant B. 2014. The Role of Game Functionality Testing in Shaping the Future of a Game. Accessed on 2 June 2019. <https://gamecloud-ltd.com/the-role-of-game-functionality-testing-in-shaping-the-future-of-a-game/>

Turunen, Tommi. 2019. Epic Games requests community help with Fortnite audio bugs. Accessed 26 May 2019. <https://fortniteintel.com/epic-games-requests-community-help-with-fortnite-audio-bugs/15833>

Vasilis. 2015. Game defect types and triggers. Accessed on 26 May 2019. <http://gamesdevelopmentvasilis.blogspot.com/2015/03/game-defect-types-and-triggers.html>

Figures

Rose, Matthew. 2014. Who Exactly Makes Games Anyway? <http://www.gameschool-prep.com/who-exactly-makes-games-anyway/>

Chandler, Heather Maxwell and Chandler, Raphael. 2011. Fundamentals of Game Development. Sudbury, MA Figure 2

Schultz, Charles P., Bryant, Robert, Langdell, Tim. 2005. Game Testing All in One. Boston, MA. Figures 3 and 4

Apex Legends. 2019. Image taken from game Apex Legends by Respawn Entertainment. Figure 5

Juego Studios. 2018. Process. Accessed on 30 May 2019. <https://www.juegostudio.com/process> Figure 6

Figure 7. Survey. 2019. Importance of QC in Start-Ups Chart.

QATestLabs. 2017. How We Test. Accessed 31 May 2019. <http://qatestlab.com/solutions/by-focus-area/games/> Figure 8

TforTesting. 2012. Test Cases for Games Apps / Checklist for Games Apps. Accessed on 1 June 2019. <https://tfortesting.wordpress.com/2012/10/04/test-cases-for-games-apps-checklist-for-games-apps/> Figure 9

Novak, Jeannie. 2012b. Game Development Essentials Game QA and Testing. Clifton Park, NY Figure 10

Bickler, Tucker. 2016. Writing the perfect bug report with JIRA. Accessed 1 June 2019. <https://medium.com/all-things-picardy/writing-the-perfect-bug-report-with-jira-26b8e4284a9f> Figure 11

Airbrake. 2011. Application Monitoring for your Entire Stack. Accessed on 2 June 2019. <https://airbrake.io/> Figure 12

Figure 13. Survey. 2019. Results of Automated Testing from Survey.

Glacier Games. 2013. [Released] NeverClip! Removes common clipping bugs! Accessed on 2 June 2019. <https://forum.unity.com/threads/released-neverclip-removes-common-clipping-bugs.276601/> Figure 14

Burke, Steve. 2015. Screen Tearing. Accessed on 2 June 2019. <https://www.gamernexus.net/dictionary/7-game-graphics-settings/39-screen-tearing> Figure 15

Software Testing Help. 2019. Defect Severity and Priority in Testing with Examples and Difference. Accessed on 3 June 2019. <https://www.softwaretestinghelp.com/how-to-set-defect-priority-and-severity-with-defect-triage-process/> Figure 16

Survey questions

Link to the survey <https://www.surveymonkey.com/r/ST6D9PP>

- What is your name? (optional)
- What company do you work in? (Optional)
- What is your role in the company?
- Why do you think testing is an important part of game development?
- At which phase of game development is testing most crucial?
- For a startup game company with not a lot of funds, how much do you think game testers are required/should be invested in? (On a scale of 0-10)
- Please elaborate with a sentence or two as to what the reasoning was behind the answer to the previous question.

- Is automated testing used often in the gaming industry/your workplace? (Yes, No, Both, No answer)
- In your opinion, should testing how fun the game is an objective for a game tester? How much importance do you think a game tester should put into testing the fun factor of a game?
- Would you like to add any other opinions on the importance of game testers/ quality control/ quality assurance in game development?

