UI-based, NPC-guided or No Tutorial at All? A Qualitative Study Comparing the Effects of Different Tutorials in the Same Game

By: Alexander Malthed & Mathias Hjelmqvist

Supervisor: Mikolaj Dymek
Södertörn University | Department of Natural Sciences, Technology and Environmental Studies
Bachelor's Thesis 30 hp
Media Technology | HT2019/VT2020
Computer Games Programme

SÖDERTÖRNS HÖGSKOLA | STOCKHOLM
sh.se

Text instruktioner, exempel eller inget alls?
En kvalitativ studie som jämför effekterna
av olika tutorials i samma spel.

Abstract

One of the many challenges of game development is creating an effective tutorial. Although tutorials can be designed to instruct in numerous different ways, little is known about how they differ in terms of effectiveness. To examine the effectiveness of the three different tutorial types, a first-person puzzle game with three versions was developed. For the purpose of an observational analysis, four participants per version (n = 12) had their gameplay recorded and were then interviewed after their gameplay session concluded. The results show that text instructions were more effective than using NPCs as examples to show the game mechanics. The lack of a tutorial made the participants vary in their understanding of the game's mechanics, some of which failed completely to learn an important game mechanic, which makes it difficult for developers to balance the later challenges in the game.

Keywords: tutorial, complexity, game mechanics, game development, qualitative study

Abstrakt

En av de många utmaningarna av spelutveckling är att skapa en effektiv tutorial. Även om en tutorial kan bli designad att instruera på flertalet olika sätt, saknas det kunskap om hur de skiljer sig i deras effektivitet. För att undersöka effektiviteten av tre olika typer av tutorials, utvecklades ett pusslespel i förstapersonsperspektiv med tre versioner. För syftet av en observationsanalys spelades spelsessionerna av fyra deltagare per version (n = 12) in, och deltagarna intervjuades direkt efteråt. Resultatet visar att textinstruktioner var mer effektivt än att använda NPCer som exempel för att visa spelmekanikerna. Saknaden av en tutorial gjorde att deltagarna varierade i deras förståelse av spelets mekaniker, varav några misslyckades helt att lära sig en viktig spelmekanik, vilket gör det svårt för utvecklare att balansera de senare utmaningarna i spelet.

Keywords: tutorial, komplexitet, spelmekaniker, spelutveckling, kvalitativ studie

Table of Contents

Abstract **3**

Abstrakt **3**

Introduction **6**

Related research **6**

Different kinds of tutorials 7

Difficulty and Complexity 9

Principles for facilitating learning 11

Research question **12**

Methods **14**

Data gathering 14

Game development 14

Complexity in Game Mechanics 16

Development Guidelines 17

Participants 18

Data handling 19

Interview questions 19

Data analysis 19

Keypoints 21

Results **24**

Observation data 24

Themes and codes from observation data 25

Frequency 28

Keypoint scores 31

Interview answers 35

Level design 35

Game mechanics 36

Instructions 37

Discussion **38**

Concepts and game mechanics that were easy to understand 38

Concepts and game mechanics that were difficult to understand 39

Tutorial types' effectiveness and improvements for further studies 40

Conclusions **44**

Appendices **48**

Appendix #1 48

Appendix #2 49

Introduction

Tutorials are a big part of many games, being used in different forms from manuals in board games to instructions on arcade cabinets, and NPCs showing the player how to use a game mechanic in video games.

For many games a tutorial can be essential, being the part that either makes or breaks the game if it proves too complex (Andersen et al., 2012, p.67). A tutorial is the new players' guidance into the designed experience, which if not done effectively might cause the player to feel lost and confused, potentially leading the player towards leaving the game in sheer frustration (Hodent, 2014, p.8).

It is therefore important to research different types of tutorials' effectiveness in an attempt to help developers choose which type to implement into their game. This will be done by developing a game that implements three different types of tutorials, and examining their effectiveness at instructing players to use the game mechanics in a meaningful way.

We will conclude with some thoughts on the tested tutorial types and their effectiveness, and propose a way to proceed with researching them further.

Related research

In this section previous research about tutorials in games and various methods that are said to help players learn the game's mechanics are summarized. Three overarching themes covering the topic could be identified from the literature.

Different kinds of tutorials

Tutorials in games is one way for game developers to teach the players the rules of the game, how the game's mechanics work and how to use them. Green et al. (2018, p.2) could identify three tutorial types commonly used in games:

- **Teaching using instructions:** Giving the player information about game mechanics with text and/or images.
- **Teaching using examples:** Teaching the player the game mechanics and how to use them by having an object or a character within the game show an example of it.
- **Teaching using a carefully designed experience:** Designing the game/level so that the player has to learn the game mechanics to progress.

Instructions can be given to the player at different times and in different situations which Andersen et al. (2012, pp.60, 66-67) refers to as *context-sensitivity*. Giving instructions in a meaningful context could for example be ingame tutorials which only shows up the first time a new game mechanic is introduced in the game. The opposite would be to give the player instructions outside of the game like the arcade cabinets did, or more modern to have all of the instructions listed one after another in an options menu.

Whittinghill and Herring (2017) compared player performance in two versions of a fighting game. In one version the tutorial only showed images of how to perform moves whereas the other one also had text. Their results indicate that the players of the version without text actually performed better in all measured aspects (ibid., p.14). Whittinghill and Herring (ibid.) then contemplate a few different causations for their findings in their conclusions including:

> "Perhaps the very presence of version-exclusive text induced users to pay unduly high attention to it, at the expense of missing animated button cues that may have been more intuitive in a quick-paced environment."
>
> "Perhaps the issue is attentional, with the text acting as a distraction, dividing users' attention between different areas of the screen when video demonstration running time was limited (especially in real time on the second and third plays)."

Text instructions could therefore benefit from being more aggressive, by perhaps pausing the game while appearing in the middle of the screen so that nothing else is missed, and to best be seen. Desurvire and Wiberg (2015, p.181) similarly found that instructions in text and audio were often missed because the player focused on something else, which could be the same reason for the participants' performances in Whittinghill and Herring (2017). As a solution Desurvire and Wiberg (2015, p.181) mean that instructions need to be repeatable and easily accessible so that the players can get the information again should they have missed it, or when they need to analyze it further in general.

The second tutorial type, teaching using example, utilizes other objects or characters in the game world to guide the player through the process of using a game mechanic and thus also showing the results of that specific action (Green 2018, p.2). Green et al. (ibid.) gives an example where a non-player character (NPC) in Megaman X teaches the player how the charging skill in the game can and should be used. Another example of using an NPC to guide the player is given by Desurvire and Wiberg (2015, p.177) where they explain that players of a certain shooter game were unable to win against the computer as the players did not realize their character could do the combination moves required to win:

> "Having AI Non-Player Characters doing these moves would demonstrate to the player that this is both possible and an option. Furthermore, demonstrating the controller buttons and thumbsticks using a controller

image and an increased health meter would demonstrate exactly how to do this."

For example, to teach the player how to move as the player character, an NPC could be used as an navigational guide for the player to follow (Moura and El-Nasr, 2014, p.14). It would move through critical points in a level while the player gets time to become comfortable with the movement controls. In addition to this, the NPC could also hint the player of where to go, or teach the player how to perform certain game mechanics and in what context they should be used (ibid., p.12). However, Moura and El-Nasr (2014, p.15) points out that overusing guiding characters can have a negative impact on the player's experience, although without specifying in what way.

The third tutorial type, teaching using a carefully designed experience, lacks an obvious tutorial since it only teaches the player by design. Andersen et al. (2012, pp.60, 67) compared freedom in tutorials, which is the degree of which the tutorials forces the player to follow the given instructions, and found that players learned better from experiencing something themselves than from reading instructions about it.

Difficulty and Complexity

As previously mentioned Andersen et al. (2012, p.63) refers to the technique of presenting guidance at the exact needed moments as context-sensitivity, which allows the player to learn the controls and character's abilities at a comprehensible pace. This also avoids overloading the player with too much information which could be discouraging. Since their results (ibid., p.67) show that tutorials have greater value in complex games and close to no impact at all in simple games the game developed for this study is therefore designed to feature complex game mechanics and difficult challenges.

When designing a difficult game with complex game mechanics that the player will have to learn, Hodent (2014, p.8) advises to balance the difficulty of the game as to facilitate the players' ability to enter a state of *Flow*. Hodent (ibid.) describes flow as "the optimal experience where the player learns best", and found that the players should be presented

with activities that are not too easy, or they will cause boredom, and yet not too difficult, risking anxiety and frustration. The game *Getting Over It with Bennett Foddy* (Foddy, 2017) is a perfect example of a game quickly losing new players possibly due to its monumental difficulty alone (React, 2017). For the goal of this study to examine different tutorials, flow becomes an interesting variable for comparison since it's advantageous for the tutorials to have the player in an optimal experience where they learn best.

Hodent (ibid.) mentions that flow is not maintained if the challenge presented to the player proves too difficult which can cause the player to become stuck on said challenge. The player might be stuck for the reason of using a game mechanic incorrectly, or that the player perceives the environment and its characteristics to not allow for the use of a specific action at all, which Linderoth (2010, p.2) calls Affordances. Having a valuable item on the other side of a drop may be one way to encourage the player to use the jump mechanic, but can the designer be sure that the player is even aware that the character has the ability to make it across? Depending on how familiar the player is with the game's mechanics and the character's abilities, the player might instead go look for an object to use as a bridge, look for another way around or simply admit the necessary character ability has yet to be unlocked. Linderoth (ibid.) explains that "The main idea is that an environment with buildings, nature, different objects, humans and animals offers the individual different ways of acting.". He continues to explain that the affordances offered by an environment is relative to whatever species or individual that is being referred to, which in this case would be the participants playing the game. Linderoth (ibid.) mentions that "Water affords breathing for a fish, but not for a human.", and that "A chair affords sitting for an adult but not for an infant.", to explain how affordances are not solely based on the species, but on the individual's expertise as well. This goes beyond the difference between an infant and an adult as well, as Gibson and Pick (2000 cited in Linderoth, 2010, p.2) states:

> "A three-inch-wide beam affords performing back flips for a gymnast, but the affordance is not realizable by others; rock climbers learn to use

certain terrains for support that do not appear to others to provide a surface of support."

Put in Gibson and Pick's terms from the quote above, depending on how well the instructions explain the game's mechanics to the player the player might end up as an "other" on a rock climbing level, without a clue on how to proceed. In other terms, the player would lack the needed affordances to climb the wall as they would not not be familiar enough with the required game mechanic.

Principles for facilitating learning

Gee (2007) discusses the player's learning process and describes several different learning principles. These learning principles can be connected to tutorials in order to better understand how to design and implement them into games.

The first one is the *Subset Principle* (ibid., p.137) which is to offer new players a simplified version of the regular gameplay in the beginning of the game, which could be seen as a tutorial area. This allows the player to learn the basics of the game while still being in the same context as the more elaborate gameplay that is to come. Within a simplified domain the game can either let the players explore for themselves or help them by having them follow a tutorial. Andersen et al. (2012, p.60), on the other hand, means that it does not need to be completely one or the other, the degree of freedom could vary between games or different parts of the tutorial itself.

However, Gee (2007, p.138) discourages tutorials with overt telling and instead chooses to emphasize discovery, letting the player experiment with the game's mechanics for themselves, which he calls the *Discovery Principle*. This should be a good principle as Andersen et al. (2012, p.67) found no evidence supporting that restricting the player's freedom would help with learning, but stating that further examination is necessary to rule out if it could improve player engagement in some cases.

When the game presents the player with a new game mechanic, Gee (2007, p.137) claims that the game should concentrate signs and actions to get the player to practice enough to

learn the new mechanic which he calls the *Concentrated Sample Principle*. Similarly, William (2009, p.8) advises developers to give players enough time to learn a concept before introducing something new.

However, as much as a level should focus on teaching the new game mechanic it is wise to still require the player to use the previously learned abilities, and even blend them together for an even more interesting gameplay experience. This is what Gee (2007, p.138) calls the *Transfer Principle*. Having challenges follow the Transfer Principle could probably also be used together with the *Incremental Principle* (ibid., p.137) which emphasizes that:

> "Learning situations are ordered in the early stages so that earlier cases lead to generalizations that are fruitful for later cases. When learners face more complex cases later, the learning space (the number and type of guesses the learner can make) is constrained by the sorts of fruitful patterns or generalizations the learner has found earlier."

The last principle is the *Explicit information on-demand and just-in-time principle* (ibid., p.138) which is that the player should be given information “when the learner needs it or just at the point where the information can best be understood and used in practice”. This concept is similar to the previously discussed context-sensitivity (Andersen et al., 2012, p.63) in that they both advise showing instructions in a specific context and time that is most relevant for the player.

Research question

From the literature review multiple articles (Andersen et al., 2012; Whittinghill and Herring, 2017; Desurvire and Wiberg, 2015; Hodent, 2014;) were identified which compared the effect of different implementations of the three tutorial types identified by Green et al. (2018): *Text Instruction*, *NPC-Example* and *Carefully Designed Experience*. However, no article was found that compared any kind of effect between the different tutorial types. This study will therefore compare the three previously covered tutorial types in terms of their effectiveness at instructing the player to use the game mechanics in a correct and meaningful way. The criteria for what makes an action meaningful in this study will be explained in *Data Analysis* as the game itself needs to be introduced beforehand.

The research question is as follows: *How effectively do the three tutorial types identified by established previous research instruct the player how to use game mechanics in meaningful ways?*

The purpose of this study is to make it easier for game developers to choose which type of tutorial to use in their game. This will be done by implementing the tutorial types into a single game in order to examine how the gameplay differs between the three versions of the game.

Methods

Twelve participants were observed and interviewed in an observational study (Creswell, 2018, p.181) while playing Afterimage, a PC video game developed by the authors for this study following development guidelines for increased learning capabilities (Gee, 2007; William, 2009; Andersen et al, 2012; Hodent, 2014; Linderoth, 2010). The data was then analysed by coding it and categorizing the codes into themes (Creswell, 2018, p.185).

Data gathering

Qualitative observations (Creswell, 2018, p.181) were used as the primary data gathering method in this study since the intention of the study is to examine player behaviour, the gameplay was recorded for coding (ibid., pp.185-186). Creswell's *complete observer* option (ibid., p.179) was chosen to avoid the researchers disturbing the participant during play. Short semistructed interviews (ibid., p.182) were also conducted using *face-to-face* or *internet* techniques (ibid., p.179) for accessibility reasons to complement the data from the observations (ibid., p.175). The interview questions regarded the difficulty of the game to better understand what the participants thought (ibid., p.181) was easy and hard.

Game development

Before any data could be collected a game containing all three tutorial types was needed, both in order to be able to compare the three tutorial types and also to make sure the participants hadn't played the game beforehand. Therefore the authors of the study made a game called *Afterimage*, a singleplayer 3D puzzle game with three different versions, one for each tutorial type, in the *Unity* (Unity Technologies, 2019) game engine.

In version A animations were implemented to play at certain points in the game in order to try to teach the player by showing examples. Version B contains text instructions without any images which shows up when the player walks a bit into a level and version

C does not contain any kind of visible tutorial at all and therefore only relies on the level design and player exploration.

Since Desurvire and Wiberg (2015, p.181) found that instructions often could be missed both the examples in version A and the text instructions in version B were made repeatable, by making the tab key show the example or instructions again.

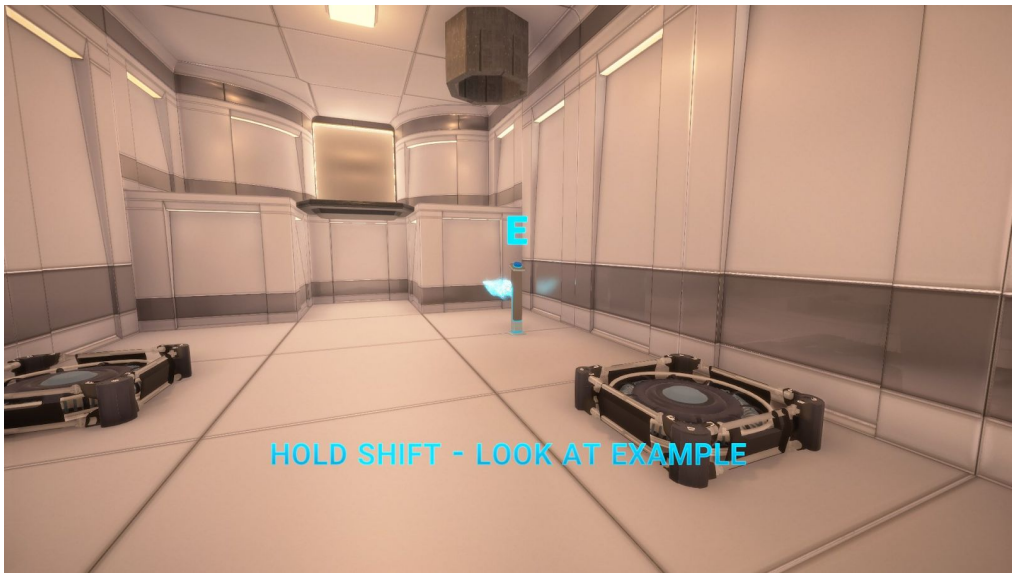

Figure 1: Example in version A's first level.

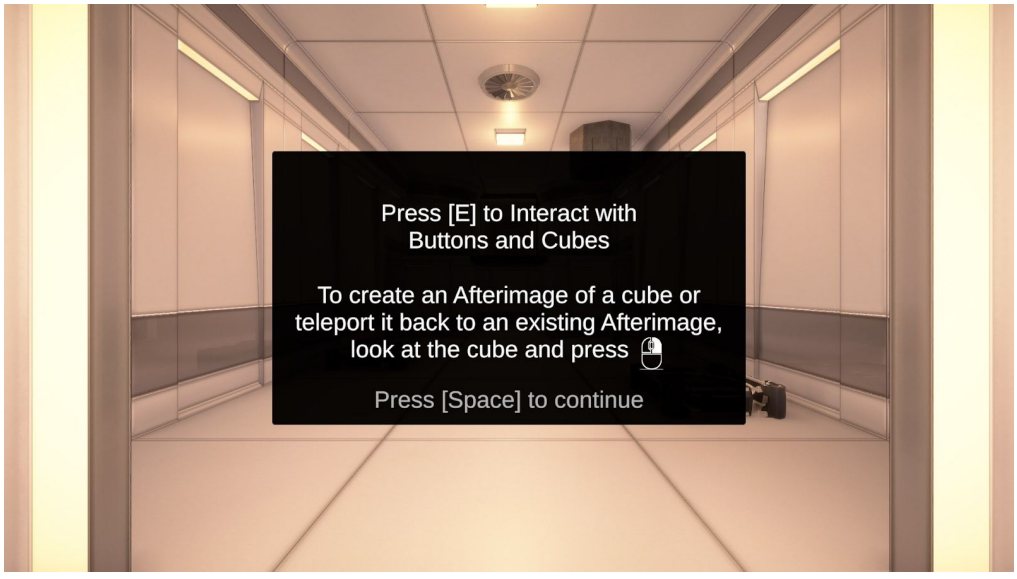

Figure 2: Instructions in version B's first level.

To hasten the process of becoming accustomed to the game's controls the standard for first person games were used for simple actions almost always found in games with a first person perspective; using the keys W, A, S and D to move around, spacebar to jump and the mouse to look around in the game.

Complexity in Game Mechanics

The game also contains two special actions which are more uncommon to see in other games and has several steps which would be more complex (Andersen et al., 2012, p.67) than basic movement like walking around. These actions are therefore more important to study since Andersen et al. (ibid.) found that tutorials are more important in complex games than simple games.

The first of the more complex actions is *Player Teleportation*. By pressing the left mouse button (LMB) an *afterimage* is spawned at the player's position in the direction the player is looking. When the player presses LMB again the player character teleports back to the afterimage, looking into the same direction as when the afterimage was placed, and launches forward with the same momentum the player had before teleporting back. This provides the player with a way to reach places otherwise unreachable. By pressing LMB while looking up and then jumping off an edge, gaining momentum while falling and pressing LMB again the player will teleport back and get launched up into the air.

The second special action is *Cube Teleportation* and works a lot like Player Teleportation. Pressing the right mouse button (RMB) while looking at a cube will create an afterimage of the cube, saving its position and rotation, and looking at the cube and pressing RMB again will teleport the cube back

Keys specific to certain versions of the game are tab and shift. In versions A and B pressing tab allows the player to view the instructions to that level again. Shift only has a function within version A, holding down the key makes the player camera rotate towards the NPC-example and center it in the middle of the screen, making it easier to follow.

Development Guidelines

The learning principles from Gee (2007) were used as guidelines when developing the game in order to make it easier for the participants to learn and know what to do.

The Subset Principle (ibid., p.137) is used in the two first levels by only focusing on teaching the player the two complex mechanics, Player Teleportation and Cube Teleportation, before having to use and combine them in harder levels later on.

Afterimage has freedom in the sense that the game does not pause and forcibly make the player do something for every step of the tutorial, following the Discovery Principle (ibid., p.138). The player is free to experiment with their own ideas and solutions at any time. There is however no choice in leaving the current level and to go do something different within the game, since that would be counterproductive for this study and its goal.

By dedicating entire levels towards single game mechanics, requiring the player to use the game mechanic multiple times within the same level to progress, the Concentrated Sample Principle (ibid., 137) is put to use. An example of this is in level three which requires the player to use Player Teleportation correctly three times in a row.

In the later levels the complex mechanics both has to be used which is according to the Transfer Principle (ibid., p.138). Level three introduces the detail that Player Teleportation brings along the held cube which is then required multiple times again in level four. Part one of level four introduces the Gravity Launchers, which is later reused in part two of the level. Level four also introduces the Laser Grids which forces the player to adapt how they use Player Teleportation which is one way mechanics have to be used in incrementally more challenging ways which the Incremental Principle (ibid., p.137) emphasizes.

The last principle, the Explicit Information On-Demand and Just-in-Time Principle (ibid., p. 138), has been used by only instructing the player of a new game mechanic when it is first needed. Instructions can also be replayed which fulfills the “On-Demand” part.

The instructions are a tool for teaching the game's mechanics to the player, but that is no use if the design on the level does not encourage the use of the said game mechanics. Hodent (2014, p.7) mentions usable level design and gives an example of such design:

> "One usable level design solution would be to place a shiny and attractive object right after a hole, such that the player has to jump over the hole to get the interesting object."

An example of this in Afterimage is in level four where a cube continually bounces back and forth between two Gravity Launchers above a deep pit, while the third Gravity Launcher is placed to cut across the cube's flight path. The cube is the shiny and attractive object in this case while the Gravity Launcher is the jump mechanic. This is an example of luring the player with a reward, but the opposite thing could be done as well where for example the Laser Grids in level four discourages the player to gain velocity in the previously learnt way.

Participants

This study used four participants per tutorial type (n = 12) in order to be able to compare any differences between the different versions. The participants were all invited through convenience sampling; acquaintances of the authors were contacted, they all had experience playing video games, were all over 18 years old and in their early twenties. The reason why acquaintances with game experience were contacted to participate was to have participants with at least basic knowledge and understanding of video games.

To test the effectiveness of the different tutorial types twelve participants played through the game or up to a maximum time of 30 minutes while being recorded. The game being a singleplayer game means that each participant was able to play on their own and did not have anyone to discuss with during the game session, leaving everything up to the participant to figure it out themselves.

Before playing, the participants were informed of how to move, jump and look around within the game. They were also informed of what other keys had a function such as

LMB, RMB and the E button, but not what function they served. This was done so that the participants would have an easier time to jump directly into the game; so that they could focus on solving the puzzle and figuring out the game's mechanics.

Data handling

All recorded gameplay was destroyed once the study concluded, and the participants' names remain anonymized to protect the participants. The participants' names are instead coded with a number and a letter; the number from one to twelve signifies which participant it is and the letter is for which game version the participant played: A, B or C. For example, the last participant of the study played version C and was therefore coded as participant 12C.

Interview questions

After each game session concluded either by completing the game or by the time running out the participants were asked two questions about the difficulty of the game:

- Was there anything in the game that was difficult to understand? What?
- Was there anything in the game that was easy to understand? What?

The first question aims to highlight those parts of the game that the participants felt that they needed more explaining from the game whereas the second question instead aims to highlight the parts of the game that the participant felt that the game explained well.

For participants playing the versions A or B, an additional question was asked.

- What do you think of the game's instructions?

The purpose of this question was to potentially gather interesting data on how these instructions could be changed and improved upon; what they did well and how or what they failed to inform and instruct.

Data analysis

To analyze the observation data as well as the interview answers the *linear, hierarchical approach* suggested by Creswell (2018, p.185) was used. This approach takes the raw

data — which in this study is the recorded videos and interview answers — and first organizes it in order to prepare it for analysis, and allows the researcher to read it all through, familiarizing with it. The following steps include coding the data which "[...] is the process of organizing the material into chunks or segments of text before bringing meaning to information" (Rossman & Rallis, 1998 cited in Creswell, 2018, p.186), and then to categorize the codes into so called themes. These steps may be repeated and data recoded until a final decision on the abbreviation for each category is made, and the codes alphabetized. Lastly, the researchers interpret the meaning of themes and descriptions for discussion.

To answer the research question the data needs to be analysed in a way that would indicate if the instructions given to the player made the player use the game mechanics in a more meaningful or insignificant way. The exact criteria for each action counting as meaningful is discovered along with the emerging codes. However, as a general guideline a meaningful action in Afterimage would help the player progress closer to the level's solution. For example to move between floors, to open a door or to acquire a cube. A game mechanic used in the intended way will count as meaningful use even if it failed. An example of this is if a participant uses Player Teleportation to launch towards a platform higher up but falls just short of reaching it.

Observations showed that participants did more or less the same things in all three versions, only a different number of times. This proves the need for another perspective when analyzing the data in order to indicate the tutorials' effectiveness. Not only was it necessary to note what the participants did, but also how many times they did each thing as well.

The number of times — the *frequency* (Trappey et al., 2005, p.1) — the more complex game mechanics are used in a meaningful or insignificant way will be recorded in order to compare the different tutorial types. This is similar to how Trappey et al. (ibid.) uses Gremler's (2004, pp.65-89) *Critical Incident Technique* in their qualitative survey to create a summary sheet describing the frequency of certain good and bad incidents.

In addition to this, certain conditions called *Keypoints* in the gameplay were planned beforehand to be used as indicators of the player's progress through each level.

Keypoints

To be able to compare the effectiveness of the different tutorial types some keypoints were planned out to look for in the recorded gameplay of the participants where the player made some kind of expected progress. For example, the first keypoint the player would finish in the game is "The player reaches the door in level one and the door is open.". These are not necessarily all placed at the end of each level, but rather at crucial points where the player needs to have understood the game mechanic or at least found a solution to that level and made progress.

The times it took the participants to get to the keypoints from the start of the level, or from the previous keypoint for keypoint 5, was recorded to be able to compare the times between different tutorial types. This could for example look like *"Player starts level X at: 15:10, Player reaches keypoint Y at: 17:23, Time spent on keypoint Y: 02:13"*.

- Keypoint 1: The player goes up the elevator and the door is open. (Level 1)

The purpose of level one is to teach the player about Cube Teleportation. The idea is for the player to save the cube's position on the button that opens the door, then place the same cube on the other button that lowers the elevator, go stand on the elevator and teleport the cube back. Doing this results in the elevator taking the player up to the opening door.

- Keypoint 2: The player reaches the second platform. (Level 2)

Level two then focuses on teaching Player Teleportation. The idea of this level is that the player will look up, save position, jump down a ledge and teleport back immediately before reaching the ground, launching the player upwards to reach the next platform.

Since the first time a player reaches a platform might be confusing for the player, the keypoint was placed at the second platform, ensuring that the player has fully understood the game mechanic.

- Keypoint 3: The player reaches the top room with the cube in hand. (Level 3)

Level three is the first level where the player has to combine the two previously learned game mechanics Player Teleportation and Cube Teleportation. Alongside that a new detail about player teleportation takes the center stage for this keypoint. This detail is that if the player teleports while holding a cube, that cube will come along to the new position as well. The only way to reach the top floor together with a cube is to jump down after already having been there once and teleport back up, bringing the cube along in that teleport.

- Keypoint 4: The player opens the door in the first room. (Level 4)

Up to this point the player has been taught that to launch from a saved player position, one must find a ledge to jump from. Level four however entirely refocuses this behavior by using *Gravity Launchers* instead, which is a mechanism that when stepped upon launches the player in the direction the launcher is facing. On top of this, lasers are blocking the player from jumping off any ledge in the level, forcing the player to come up with a solution using the launchers instead.

To solve the first room of level four it is planned so that the player needs to combine previously learned mechanics with the Gravity Launchers. First off the player has to use the Gravity Launcher to fly across the room, grab the cube and teleport back to safety. Although it is possible to drop or throw the cube onto the button the idea is for the player to save the cube's position above the button that opens the door, teleport back while still holding on to the cube and then teleport the cube back so that it falls down onto the button. Reaching keypoint four means that the player should have an understanding of how the Gravity Launchers work and how they can be combined with other mechanics.

- Keypoint 5: The player opens the door in the second room. (Level 4)

The solution to the second room is quite similar to the first room of the level in the way that it requires the player to combine multiple mechanics but also remember to use the Gravity Launcher from the previous room. Successfully completing keypoint five should show a complete or near complete understanding of the game's mechanics.

Results

Observation data

From the observation analysis emerged similar codes referring to the more complex game mechanics, but in different ways. For example, Player Teleportation was used both in a meaningful way such as to reach higher ground or to avoid a laser grid, but also in an insignificant way where the player redirects the velocity downwards when expecting to fly upwards. This stays true for the different codes related to the complex game mechanics and from that surfaced the two themes *Meaningful Use of Game Mechanic* and *Insignificant Use of Game Mechanic*. These codes occur in all of the game versions, only a different number of times, which will be referred to as their *Frequency*. For example if a version saw a great number of failed uses of game mechanics while another version only saw a mere few before seeing successful uses, that would imply the instructions of the second version were more effective. *Frequency* will be further discussed later in the *Results*, but first let's take a look at the other themes.

Additional smaller themes regarding player behaviour include *View Instructions*, *Player Tricks* and *Misconceptions*. *View Instructions* refers to whenever the participant would read or watch the instructions shown in versions A and B. Player Tricks on the other hand includes codes that arose whenever the player would find a surprising, not thought of way to progress in the game. This includes a player figuring out how to throw a cube and use that to open a door, as well as a player solving a level without using the intended game mechanic at all. Lastly, the Misconceptions theme categorizes all codes that refers to incorrect assumptions by the players regarding the game mechanics. For example how some players assumed there was a range limit of how far away one could be when teleporting a cube, when in fact, there is no such limit and the instructions failed to inform the players of this.

Themes and codes from observation data

The bullet point list below shows the different themes and codes that emerged from the observation analysis. Each code has one or more criterias for a player action to count as belonging to that code. These criterias were discovered alongside the emerging codes as the data was being analyzed, and are shown in the parentheses following the code. Only one criteria needs to be satisfied for the player action to count as a meaningful use, if none are satisfied the action is deemed insignificant.

Before showing the list it is important to explain its structure. The presentation of these themes and codes follows the structure shown below where the themes are listed with their underlying codes shown indented beneath them. Each code also lists their criteria in their following parentheses and then explains them in order.

Here is the structure:

- **Theme name**
    - **Code name related to theme above (Names of the criterias)**
      Criteria #1: *Explanation of what player action satisfies this criteria.*
      Criteria #2: *Another explanation…*

Here follows the list of themes and codes that emerged through the data analysis:

- **Meaningful Use of Game Mechanic**
    - **Player Teleportation (Avoidance, Mobility)**
      Avoidance: *The player avoids flying into a laser grid.*
      Mobility: *The player flies or moves between different floors. Even if flying too low and not reaching the destination by a small margin.*
    - **Cube Teleportation (Acquisition, Activation/Deactivation)**
      Acquisition: *The player reaches a cube with the use of RMB.*

Activation/Deactivation: *The player uses RMB to activate or deactivate an activator.*

- **Player Teleportation with Cube (Acquisition, Transportation)**
  Acquisition: *The player acquires the cube in level 4 part 1 with the use of Player Teleportation.*
  Transportation: *Transports the cube between different floors or towards an activator.*

- **Insignificant Use of Game Mechanic**

  - **Player Teleportation (Minor or No Movement)**
    Minor or No Movement: *Player redirects velocity towards the ground when expecting to fly upwards. Player teleports to a destination on the same floor as already standing on without progressing.*

  - **Cube Teleportation (Without Progress)**
    Without Progress: *Cube is teleported in a way that does not help gain progress. For example to the very same position as it currently is, or to the same floor without activating or deactivating an activator.*

  - **Player Teleportation with Cube (Minor or No Transportation Distance)**
    Minor or No Transportation Distance: *The player teleports with the cube in hand in a way that does not yield progress. For example to the same position or floor as already standing on.*

- **View Instructions**

  - **Watch examples - Version A (Auto, Tab)**
    Auto: The first time the instruction automatically shows in a level.
    Tab: Whenever the player presses tab to show the instruction again.

- **Read Instructions - Version B (Auto, Tab)**

  Auto: The first time the instruction automatically shows in a level.

  Tab: Whenever the player presses tab to show the instruction again.

- **Player Tricks**

  - **Cube Throw (Deliberate Throw)**

    Deliberate Throw: *For example, participant 5B notices that the cube is saved too low beneath the Final Door Activator in level 4 part 2. The participant then decides to throw the cube straight up and teleport the cube before it reaches the roof. This results in the cube flying up and into the activator, opening the door.*

  - **Unintended Solutions for Level One (Button, Player Teleportation)**

    Button: *Using the button to respawn the cube and quickly board the rising elevator.*

    Player Teleportation: *The player completes level one by using LMB instead of RMB.*

- **Common Misconceptions**

  - **Player Teleportation (Waiting too long)**

    Waiting too long: *Thinking more airtime gives higher velocity instead of redirecting current velocity. (Some players waited as long as they could when flying before teleporting themselves.)*

  - **Cube Teleportation (Max Range, Redirecting Velocity)**

    Max Range: *Thinking one must stand near the cube to teleport it. (Some players brought the cube along onto the elevator before teleporting it back)*

    Redirecting Velocity: *Thinking the Cube Teleportation game mechanic*

works similar to the Player Teleportation in that it redirects the velocity, which it does not.

Frequency

The number of times, or frequency, the participants on average used the more complex game mechanics in a meaningful or insignificant way was recorded for each game version. The following diagrams show each complex game mechanic's meaningful- and insignificant frequencies color coded for each level. Each diagram shows the three different game versions ordered horizontally and split into meaningful and insignificant uses. The levels of the game are color coded and ordered vertically on each game version, starting with level one in the bottom. The increasing number on the y-axis shows the frequency.

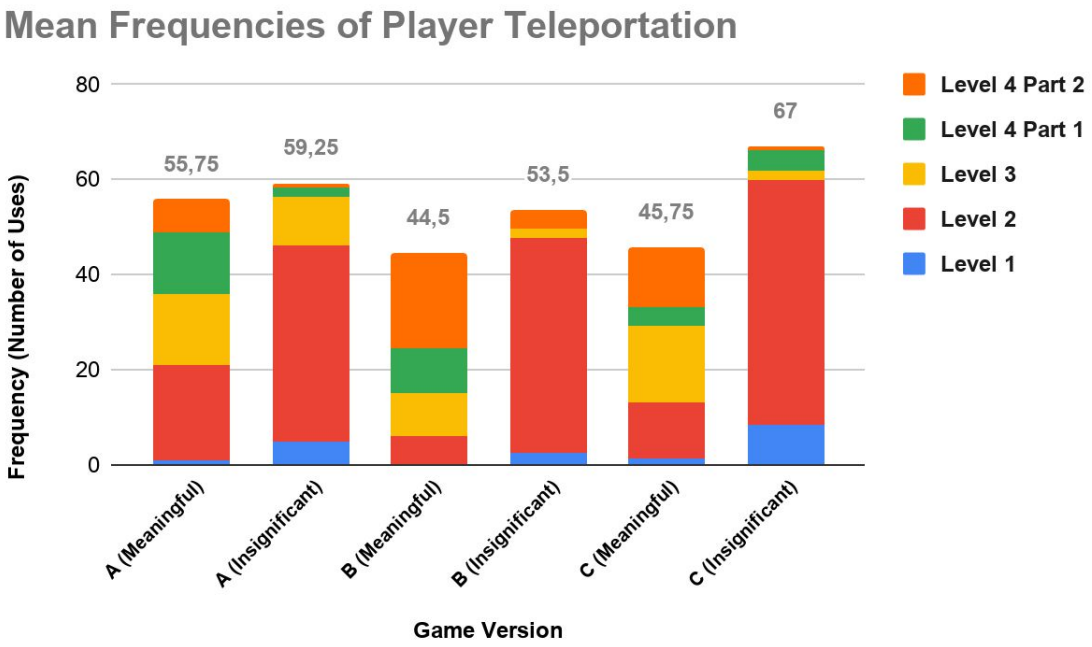

Figure 3: The mean number of times participants used the Player Teleportation game mechanic in a meaningful- or insignificant way.

The diagram shows version B to have the least number of insignificant uses as well as the least number of uses in total. Although the difference between the insignificant uses between version A and version B is small, further insight into the individual frequencies

of version B shows a deviant frequency. This number pushes the mean frequency of insignificant uses in level two of version B, from 5,66 to 45 (see Appendix #2).

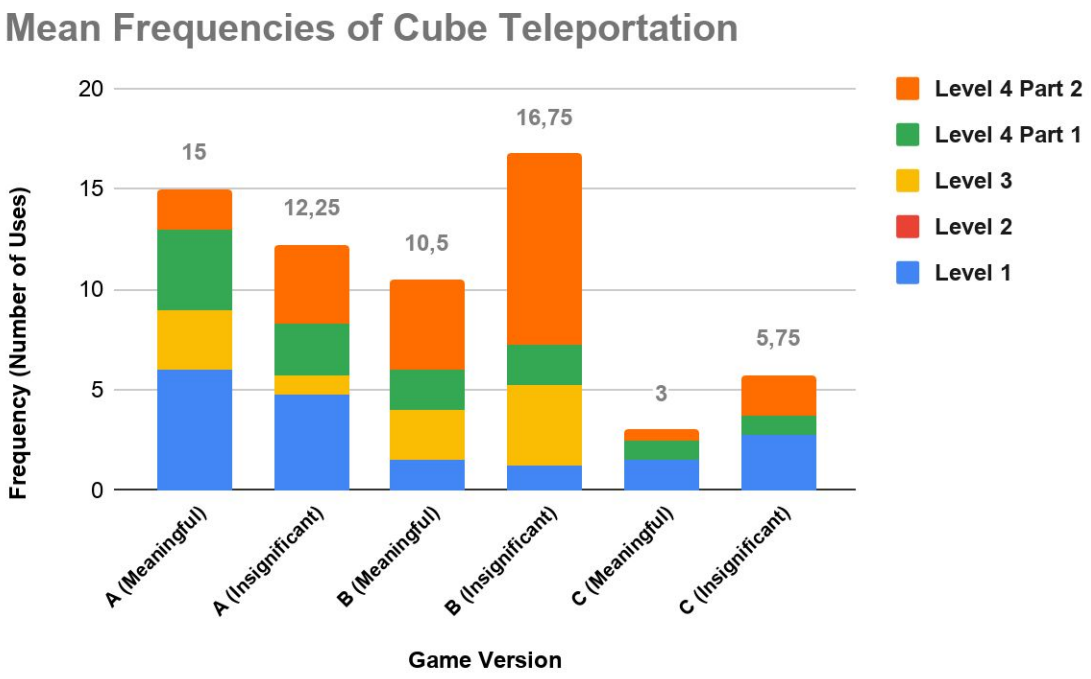

Figure 4: The mean number of times participants used the Cube Teleportation game mechanic in a meaningful- or insignificant way.

The diagram indicates that for level one, participants of version B understood the game mechanic faster than other versions, and that although they struggled with it on later levels, they used it. Version A shows a higher number of uses than other versions for level one, both insignificantly as well as meaningfully. Version C on the other hand did not use the game mechanic as much as the other versions, and most importantly not at all in level three.

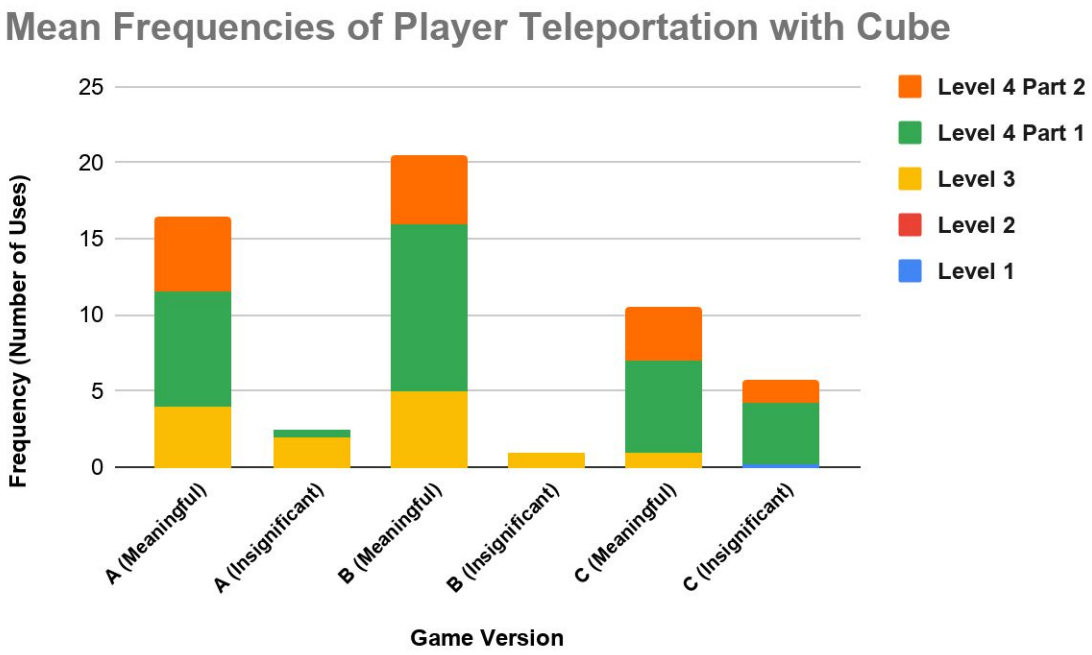

Figure 5: The mean number of times participants used the Player Teleportation game mechanic while holding a cube in a meaningful- or insignificant way.

Here it is shown how participants were quick to learn how to teleport with the cube in a meaningful way and that versions A and B were slightly faster than C. Except for level three where the participants of version C used the game mechanic correctly on their first try.

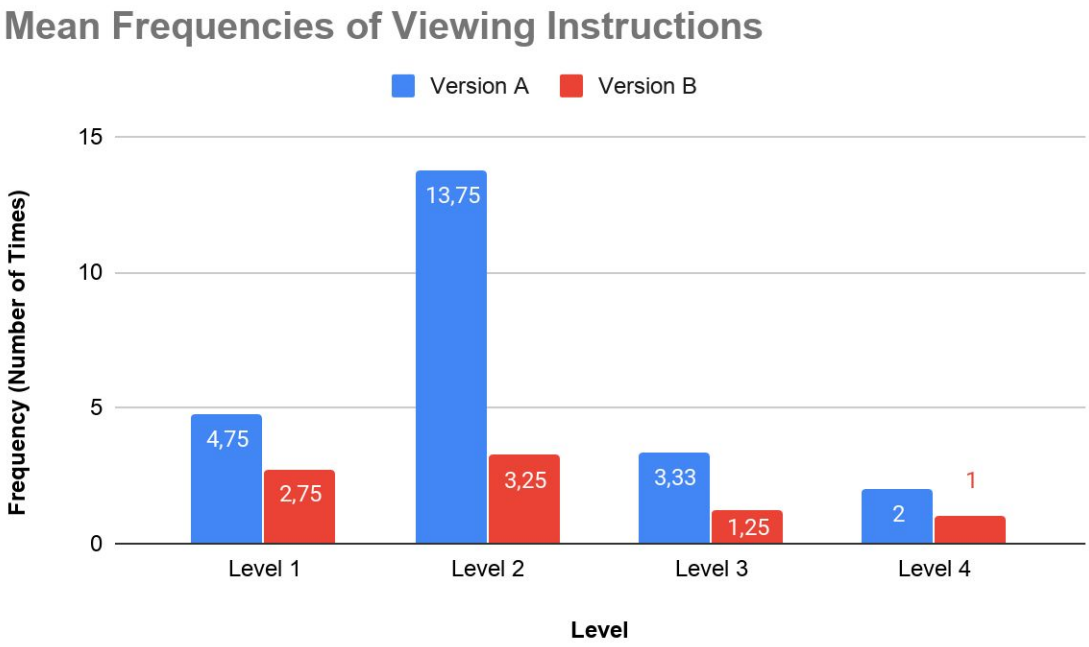

Figure 6: The mean number of times participants viewed the instructions.

The player can "lose" in level 4 and restart the level, therefore the automatically played instructions was only counted once the first time it played and then additionally when the participant replayed the instructions by pressing tab. Between the two versions with instructions participants playing version A watched the instructions more.

Keypoint scores

Each diagram shows how much time the participants spent on a specific keypoint, and is divided into the three game versions. The different players are distinguished by their version and color which remains the same throughout the different keypoints, this means that the blue player in keypoint one is also the blue player in keypoint two. It is however important to note that the blue player in version A is not the same player as the blue player in version B.

Furthermore, as the game proved to be quite challenging many of the players did not complete the game within the given time limit of 30 minutes. This is shown in the diagrams by the player's color being faded, which means that the player spent time trying to complete the level but ultimately ran out of time.

Other edge case scenarios that occurred were when the player managed to get the keypoint just in time before the end of the session, or when the player reached the keypoint but then failed to proceed to the next level. In the diagrams this is shown by a completed keypoint of full color, but no representation of that player in the following keypoint. An example of this is the red player in keypoint 4, version C.

The average time spent on a keypoint by players of a certain version was calculated using only the timings created by players who reached the keypoint. This was done in order to compare equal to equal since these players have all achieved the same goal. This way small values such as 0,3 (minutes) in keypoint four version A is not included since the player did not actually manage to complete the level in 18 seconds, just barely visited it.

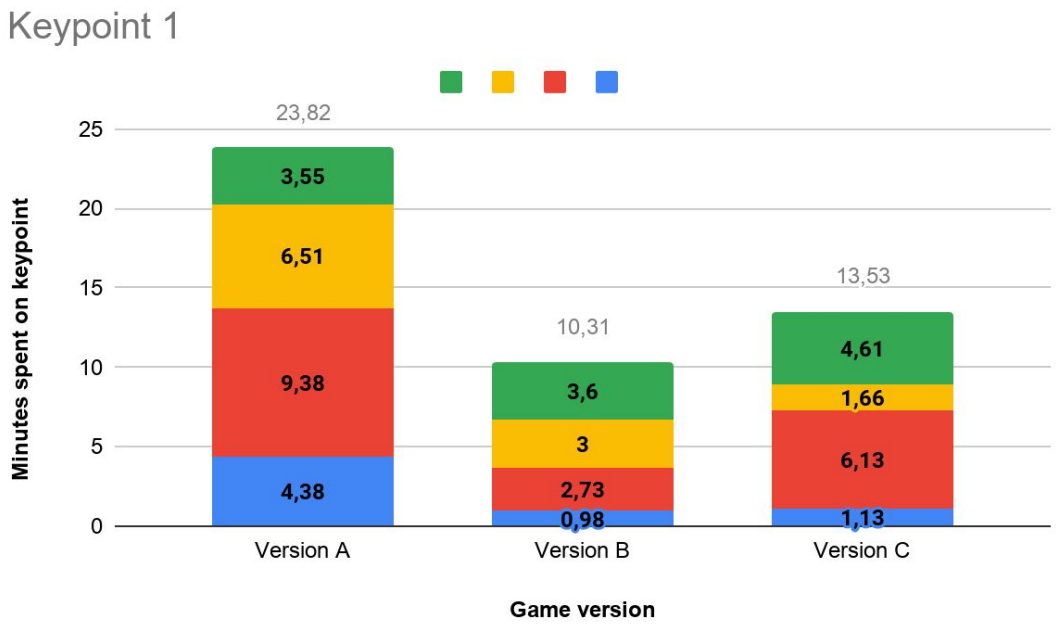

Figure 7: Mean number of minutes it took participants of the different game versions to reach the keypoint in level one.

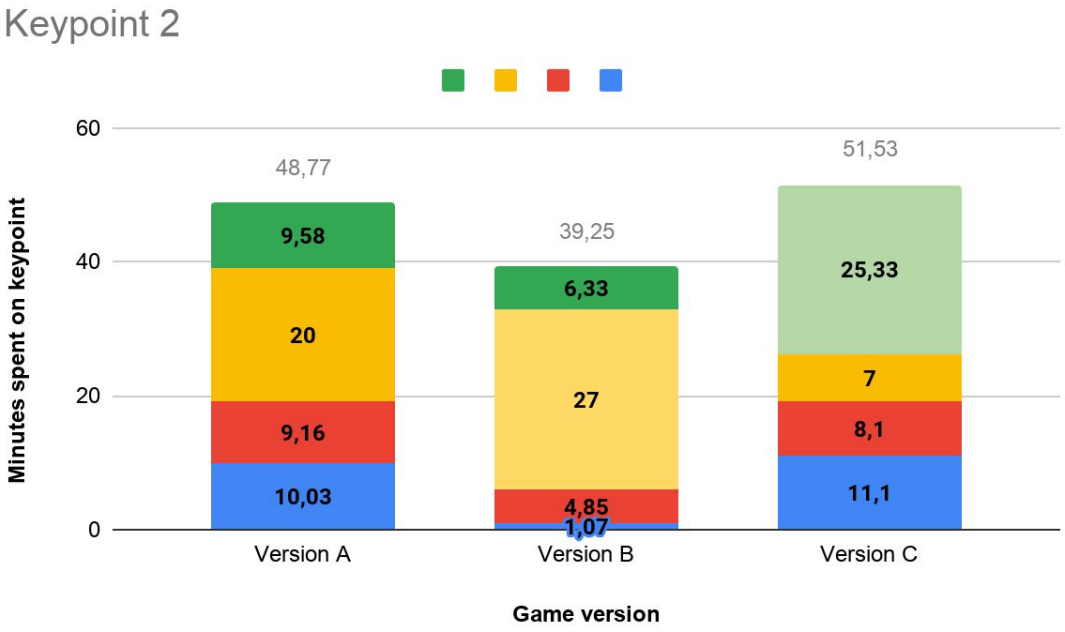

Figure 8: Mean number of minutes it took participants of the different game versions to reach the keypoint in level two.

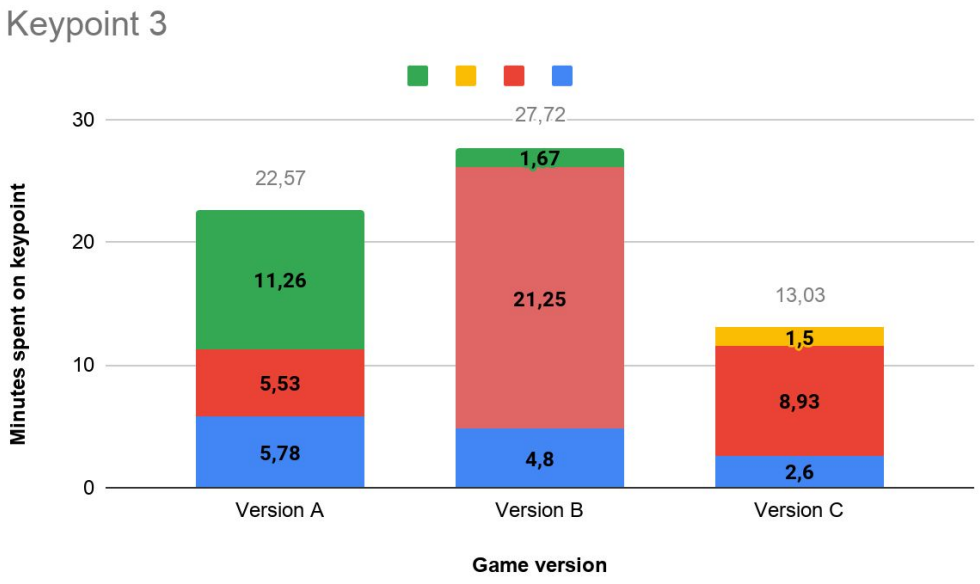

Figure 9: Mean number of minutes it took participants of the different game versions to reach the keypoint in level three.

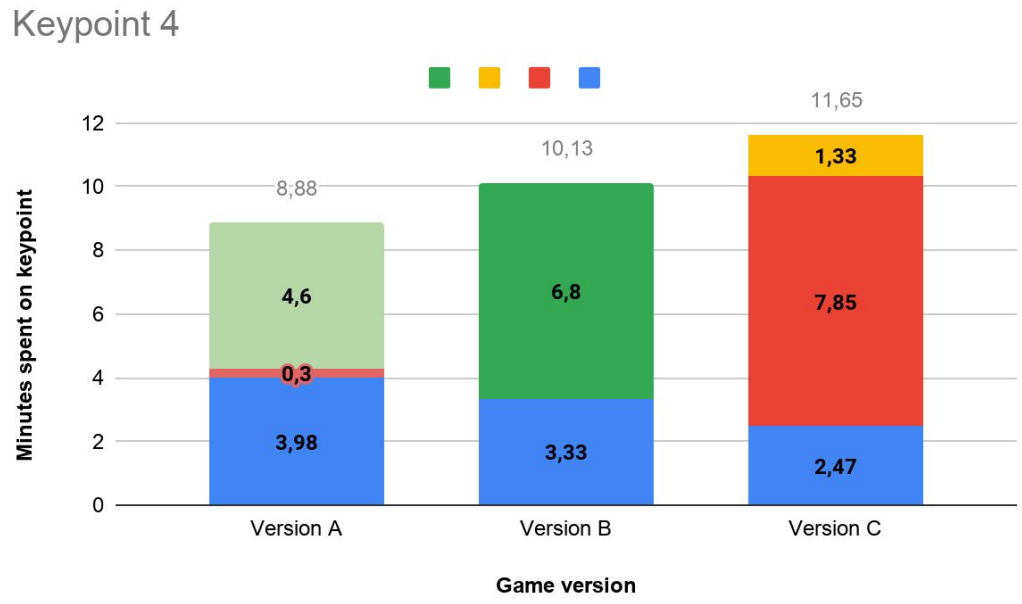

Figure 10: Mean number of minutes it took participants of the different game versions to reach the first keypoint in level four.

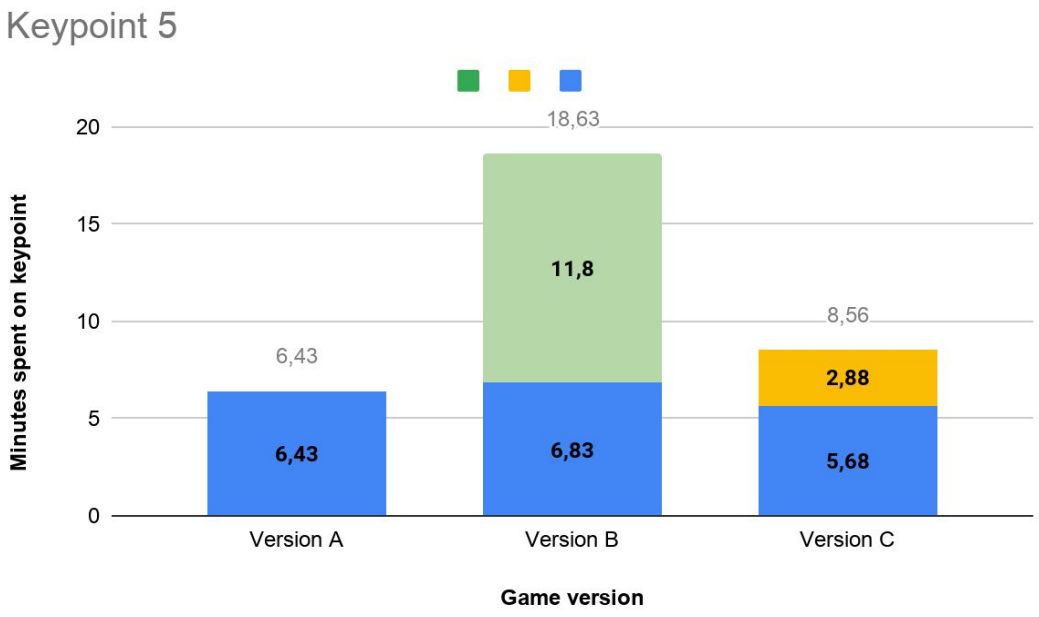

Figure 11: Mean number of minutes it took participants of the different game versions to reach the second keypoint in level four.

Interview answers

From the analysis of the interview data emerged codes regarding concepts that were easily understood, and other concepts that were more difficult to understand. These codes could then be categorized into the three overarching themes *Level design*, *Game mechanics* and *Instructions* which are all explained below.

The interviews were conducted in swedish are the answers translated to english by the authors.

Level design

The codes of level design include *Tricky Levels* and *Easy to understand the goal*. Tricky Levels were a common answer given by participants of all versions of the game where they explain that the difficulty lies not in performing the solution, but in discovering it.

> "But the tricky bit was how [the puzzle] was supposed to be done."
>
> **Participant 1A**

> " [...] the game was pretty tricky, but I anticipated that it would be."
>
> **Participant 6B**

Multiple participants also mentioned that the goal of the level was implicitly and easily understood. They instantly knew what they wanted to achieve, which was to get to the end of the level.

> "I quickly understood what I was supposed to achieve, what the puzzle was about.[...]" **Participant 1A**

> "I liked how the goal was not stated, but it was implicitly understood that you want to get to the end, so it was easy to understand." **Participant 5B**

Game mechanics

Several participants explained that the velocity redirection of the Player Teleportation game mechanic was difficult to understand, and that it specifically had to do with the detail of having to look up before saving.

> "[...] such as looking upwards before jumping didn't feel clear to me."
>
> **Participant 2A**

> "I had a little trouble figuring it out, or maybe it just took a long time for me, to realize that I must look in the direction I am to fly towards."
>
> **Participant 11C**

Other game mechanics that are said to be much easier to understand are the cubes and the activators, and how they relate to each other. Some of the participants pointed out that the result of putting a cube on an activator — an open door for example — was among the easier things to understand in the game.

> "Yes, that the cubes should be set on the "portal" things (Activators). Felt given." **Participant 2A**

> "The easiest thing to understand was that the box on the plates is equal to open." **Participant 4A**

Some of the participants also stated that the basic game mechanics were easily understood as well. This includes controls for moving around, jumping and pressing a button, with participants saying that the controls for walking and moving were as usual, and that it was easy to know the purpose for each button.

> "Otherwise, the basics of moving and likewise were easy to understand also." **Participant 5B**

> "I thought the game was easy with the controls, it was easy to know which buttons did what." **Participant 6B**

Instructions

Two of the four version A participants mentioned that it was *Difficult to understand the NPC-Example* when asked what they thought about the instructions:

> "(It was) hard to understand the blue spirits [...]" **Participant 2A**

> "They were not so obvious [...]" **Participant 4A**

On the other hand, majority of version B's participants mentioned that it was *Easy to understand the text instructions* and also that they found it *Good to be able to reread the instructions*:

> "I thought they were good in the context of that you wanted to introduce and show a concept/demo [...] especially that you could press tab to get the old instruction [...]" **Participant 6B**

> "I often have to reread the instructions as I often swap around the words or misread, so first and foremost it was very nice to be able to read the instructions again if you wanted. The instructions, I thought, were clear and good." **Participant 7B**

Discussion

From the analysis of the observation recordings, the interview answers as well as the keypoint scores, some concepts and game mechanics were identified as being more easily or difficult to understand than others. The easier ones will be covered ahead of the more difficult ones starting with the most basic mechanics in Afterimage.

Concepts and game mechanics that were easy to understand

Both the recordings and interview answers show that the participants did not have any problems at all to understand the simple game mechanics in the game like walking, jumping and picking up cubes which are commonly found in other first person games. This is in accordance with Andersen et al. (2012, p.67) who found that tutorials had a small impact on less complex games.

Another easily understood concept was the relation between a cube and an activator, how these two objects interact with one another and what purpose their interaction serves. One participant even stated that this concept was the easiest thing to understand in the game, and another one said that it felt given. The interview results showed that participants from all versions of the game shared this sentiment. It is thus safe to say the concept of cubes and activators were simple enough not to benefit from a tutorial (ibid.), and only distracted the players for a short duration if anything.

The final easily understandable concept was the navigational goal of the game, multiple participants explained how what they were supposed to achieve was implicitly and clearly understood right from the start, without any navigational advice helping them out.

These game mechanics were all the easiest understood ones which is likely because they are also the most generic game mechanics found within Afterimage, meaning participants have likely played other games that use the same controls for the same purpose. One participant explains that the controls were easy to understand because of having played similar games before that were structured in the same way. Meaning that the controls

were not only easy in and of itself, but because the player had previous experience gained from other games. In terms of Linderoth's (2010) affordances, one might say the player entered the game with some affordances already unlocked as the player realized the similarities to another game. The player was therefore better equipped to tackle the challenges faced within Afterimage.

Although Afterimage has some game mechanics simple enough to not need a tutorial, it does have other more complex ones that the results would indicate definitely require further instructions.

Concepts and game mechanics that were difficult to understand

The observations showed that the participants had the most problem with Player Teleportation. When asked if there were anything difficult in the game, more than half of the participants mentioned the velocity redirection mechanic. More specifically their answers showed that the difficulty laid in having to look in the direction the player wanted to get launched in. The scores from keypoint two also indicates that players spent most of their time in level two, dealing with the teleport mechanic.

This all points to the complexity of the Player Teleport, but the observation analysis shows that even when players understood the game mechanic and used it correctly, they would still struggle to progress in the level, indicating its difficulty as well. Interview data suggests that the struggle many of the players suffered through during level two had an impact on the players' flow (Hodent, 2014, p.8) as one participant even mentions hating the level. The more failed attempts a participant had, the more often they would try the very same approach repeatedly without altering their attempts, which could be a sign of the difficulty having disrupted the player's flow to the point of frustration.

Another source of difficulty could be the participants' own perceived difficulty of not being able to perform certain actions due to a lack of understanding the game mechanics. For example, several of the participants always tried to find a way of solving the levels in

a way that would not utilize the possibility to teleport a cube at range, which makes the levels more difficult than they actually are.

Another misconception seen from the analysis is that some participants assumed the velocity redirection of the Player Teleportation would also mean the Cube Teleportation would work in the same way. Participants would line up the cube toward an activator and fire it off with a gravity launcher only to find the cube maintaining its previous velocity.

Finally, some players would wait until the very last moment before teleporting themselves when using a gravity launcher, which often meant when most of the velocity given by the launcher has been lost to either gravity or air resistance. This could mean that the players did not know the exact functionality of the game mechanic, and perhaps thought a longer airtime meant more velocity after teleporting. These perceived difficulties are only difficult due to the participants not having unlocked the required affordances yet (Linderoth, 2010, p.2). Once the player learns how the game mechanics actually works these perceived difficulties disappear. Which is where the tutorial comes in with the purpose to inform the player of their character's abilities, unlocking new affordances for the player.

Tutorial types' effectiveness and improvements for further studies

Observations show that the participants of both versions A and B repeated the instructions when they got stuck, and that they could therefore sometimes make progress because of that. Other players repeated the instruction but still did not understand them, no matter how many times they were viewed. The participants who did not understand an instruction after viewing it three times did not understand it any better after viewing it yet another time. It would therefore be favorable to not only repeat the same instruction as advised by Desurvire and Wiberg (2015, p.181), but to build on the idea by also increasing the clarity or rearticulating the instruction after multiple views.

The level that saw the most number of repeats of the instructions is level two, featuring the Player Teleportation game mechanic. Although the frequencies for insignificant use

of Player Teleportation are not that far apart between the game versions, it is important to note that version B had a deviant case with a participant using the game mechanic insignificantly 163 times (see Appendix #2), increasing the mean frequency in level two from 5,66 to 45. Even with this deviant case version B still holds the lowest mean frequency of insignificant uses of Player Teleportation in level two of all game versions, meaning that it was the version which instructed the players about the mechanic the best.

The average time it took participants to reach keypoint two (see Appendix #1) also indicates toward version B being a more effective tutorial as the participants did not end up spending as much time in the level as participants of other versions did. This result is not restricted to level two alone, but manages to stay true over the first three levels, creating a pattern showing the effectiveness of version B.

Desurvire and Wiberg (2015, p.181) mentions that a weakness of text-based instructions is that they are easily missed by the player. However, this weakness was not found in the results of this study where every participant noticed and took their time reading through it all before proceeding. This is likely because of the use of a more aggressive approach to showing the instructions, by pausing the game and centering the text in the middle of the screen. This way the participants did not have to worry about missing any other information or event happening in the game while reading. However, it is worth noting that pausing the game in order to show an instruction may only be felicitous in a single player game to avoid disturbing any other players. Although having such positive results, version B can still be improved upon, by for example including images as shown by the results given in Whittinghill and Herring (2017, p.14).

Contrary to what the authors expected, results show version A not being as efficient as previously thought. The keypoint times shows version A being considerably slower than the other two versions which could partially be explained by the tutorial performing a visual spectacle, encouraging the player to stop and watch the NPC-Example move around. The interview results however show that the participants found it difficult to understand what the NPC-Example was trying to show. Which is also further hinted

towards by the observation analysis, showing that participants of version A repeated the instructions more times than those of version B. In addition, the observation analysis shows that on average, participants of version A would use the Player Teleportation insignificantly close to 60 times during level two before reaching level three, where a continued struggle to use it meaningfully is shown.

Players not looking up is however a fairly common problem in game design as Jeep Barnett — a developer on *Portal* (Valve corporation, 2007) — says in the game's ingame commentary:

> "One bizarre fact of game design is that, without some serious prompting, players will rarely look up. In this case, the prompt is a ladder. Most players will investigate where the ladder goes, which is up. The ladder actually falls apart as soon as it's touched, but by then it's served its purpose." (The Portal Wiki, 2014. #002)

In version A the player is instructed to look up by an image of a computer mouse moving upwards towards the destination, which the results show was too vague as none of the participants picked up on that. For further studies the NPC-Example of this study could be improved upon by using Moura and El-Nasr's (2014, pp.13–14) *Direction from Characters* technique in a more straight to the point fashion and have an NPC not only show an example of using the game mechanic, but to also directly tell the player to look up. Another more subtle way could be to display an image of an eye and an upwards pointing arrow during the NPC-Example if voice lines does not make sense in the context of the game.

As a closing point, the average amount of times Cube Teleportation was used in version C was less than the other versions (Figure 3-5), both in meaningful and insignificant ways. The reason for the low frequencies in version C could be that it is the only version that does not inform the player of the game mechanic's existence, nor how to use it.

Although Andersen et al. (2012, p.67) found that players learned more from experience than from reading instructions, in this case the players managed to learn how to teleport themselves at the wrong time and thus entirely skipping having to learn about Cube Teleportation at all. This resulted in the participants relying more on Player Teleportation for figuring out the solutions. Which worked out for them in this case, but say if a theoretical level five would have followed and required the use of Cube Teleportation for its solution, the participants of version C would have been worse off than those of other versions. It would therefore be wise to have some degree of control of what game mechanics the player learns as the player progresses through the game, when following Andersen et al.'s (ibid.) advice on letting the player learn solely from experience.

Conclusions

The three different tutorial types identified by established previous research: Text-Instructions, NPC-Example and Carefully Designed Experience, where all tested in the same environment — in the game Afterimage — and had their effectiveness discussed in terms of their observation- and interview data.

Starting with the NPC-Example, which did not prove to be as effective as previously thought. Participants repeated the NPC-Example instructions more often than the Text-Instructions, although without the participants learning anything new with the repeated views. NPC-Example was however slightly more effective at instructing the player to use the Cube Teleportation game mechanic.

Out of the three tutorials tested, Text-Instruction proved to be the most effective at instructing the player how to use the game's most complex game mechanic Player Teleportation. This is based on the fact that it had the average frequency of insignificant uses, even with a deviant case increasing the average by a margin, and also the short time it took the participants of the Text-Instruction version to complete Keypoint 2.

The version only relying on being a Carefully Designed Experience however almost completely failed in teaching the participants about Cube Teleportation, which was used meaningfully only three times on average through all levels. This could make it difficult for developers to balance the challenge of later levels requiring the use of the game mechanic as players' understanding of it will vary.

All tutorials struggled with the player not looking up before saving their position when first introduced to the Player Teleportation game mechanic. The authors therefore urge developers to follow Jeep Barnett's advice of seriously prompting the player to look up when needed, no matter what tutorial one might use.

This study used a slow paced singleplayer puzzle game as a unit of study, and recruited participants using convenience sampling. Further studies should therefore use a more

randomized sample, and examine if the tutorials used in this study yield similar or different results when implemented into other types of games, such as a fast paced action multiplayer game.

References

Andersen, E., O'Rourke, E., Liu, Y.-E., Snider, R., Lowdermilk, J., Truong, D., Cooper, S. and Popovic, Z., 2012. The Impact of Tutorials on Games of Varying Complexity. In: *CHI '12, Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. Austin, Texas, USA, 5-10 May 2012. New York: USA. pp.59-68. 10.1145/2207676.2207687.

Desurvire, H. and Wiberg, C., 2015. User Experience Design for Inexperienced Gamers: GAP – Game Approachability Principles. In: R. Bernhaupt, ed. 2015. *Game User Experience Evaluation*. Cham: Springer, pp.169-186. 10.1007/978-3-319-15985-0_8

Foddy, B., 2017. *Getting Over It with Bennett Foddy*. [Computer game] Steam: Bennett Foddy.

Gee, J, p., 2007. *What Video Games Have to Teach Us About Learning and Literacy*. 2nd ed. New York: Palgrave Macmillan.

Green, M.C., Khalifa, A., Barros, G. A. B. and Togelius, 2018. *"Press Space to Fire": Automatic Video Game Tutorial Generation*. [online] Available at: <https://arxiv.org/pdf/1805.11768.pdf> [Accessed 12 November 2019].

Gremler, D. D., 2004. The critical incident technique in service research. *Journal of Service Research*, [e-journal] 7(1), pp.65-89. https://doi.org/10.1177/1094670504266138.

Hodent, C., 2014. Toward a Playful and Usable Education. In: F. C. Blumberg, ed. 2014. *Learning by Playing: Video Gaming in Education*. New York: Oxford University Press. nationalCh.10. DOI: 10.1093/acprof:osobl/9780199896646.003.0010.

Linderoth, J., 2010. Why gamers don't learn more: An ecological approach to games as learning environments. In: *DiGRA Nordic '10, Proceedings of the 2010 International DiGRA Nordic Conference: Experiencing Games: Games, Play, and Players*, 9. University of Gothenburg. ISBN / ISNN: ISSN 2342-9666.

Moura, D. and El-Nasr, M. S., 2014. Design Techniques for Planning Navigational Systems in 3-D Video Games. *Computers in Entertainment (CIE)*, 12(2), pp.1-25. DOI: 10.1145/2701657.2633421.

React, 2017. *RAGE QUIT SIMULATOR!? | GETTING OVER IT (React: Gaming)*. [Video online] Available at: <https://www.youtube.com/watch?v=6cDIblcYDkA> [Accessed 25 November 2019].

The Portal Wiki, 2014. *Portal developer commentary.* [online] Available at: <https://theportalwiki.com/wiki/Portal_developer_commentary#.23002> [Accessed 5 Mars 2020]

Trappey, C., Hsiao, T.-T., Chang, T.-C., Che, M.-H. and Chiu, W.-J., 2005. Consumer Driven Computer Game Design. In: *DiGRA '05 - Proceedings of the 2005 DiGRA International Conference: Changing Views: Worlds in Play*, 3. ISBN / ISNN: ISSN 2342-9666.

Unity Technologies, 2019. *Unity* (2019.2). [computer program] Unity Technologies. Available at: <https://unity.com/> [Accessed 8 Mars 2020].

Whittinghill, D. and Herring, D., 2017. A Comparison of Text-Annotated vs. Purely Visio-Spatial Instructions for Video Game Input Sequences. *Computers in Entertainment (CIE)*, 15(2). 10.1145/2700527

William, R. and Martin, A. S., 2009. Evaluating Interactive Entertainment using Breakdown: Understanding Embodied Learning in Video Games. In DiGRA '09, *Proceedings of the 2009 DiGRA International Conference: Breaking New Ground: Innovation in Games, Play, Practice and Theory*, 5. Brunel University. ISBN / ISNN: ISSN 2342-9666

Appendices

Appendix #1

Chart 1: Mean time spent on keypoint one (All participants).

Version (All participants)	Mean Time (Minutes)
A	**5,96**
B	**2,58**
C	**3,38**

Chart 2: Mean time spent on keypoint two (Finishers).

Version (Finishers)	Mean Time (Minutes)
A	**12,19**
B	**4,08**
C	**8,73**

Chart 3: Mean time spent on keypoint three (Finishers).

Version (Finishers)	Mean Time (Minutes)
A	**7,52**
B	**3,24**
C	**4,34**

Chart 4: Mean time spent on keypoint four (Finishers).

Version (Finishers)	Mean Time (Minutes)
A	**3,98**
B	**5,07**
C	**3,88**

Chart 5: Mean time spent on keypoint five (Finishers).

Version (Finishers)	Mean Time (Minutes)
A	**6,43**
B	**6,83**
C	**4,2**

Appendix #2