HOCHSCHULE
MITTWEIDA
University of Applied Sciences

Faculty Applied Computer Sciences and Biosciences

Professorship Media Informatics

Bachelor Thesis

Implementation strategies for battle AI and its usage to increase replayability in action RPG games that focus on PVE combat

Tina Kießling

Mittweida, the 14th August 2022

First Examiner: Prof. Dpl.-Ing. Alexander Marbach
Second Examiner: M.Sc. Manuel Heinzig

Kießling, Tina
Implementation strategies for battle AI and its usage to increase replayability in action RPG games that focus on PVE combat
Bachelor Thesis, Faculty Applied Computer Sciences and Biosciences
Hochschule Mittweida — University of Applied Sciences, August 2022

Abstract

The aim of this bachelor thesis is to find out how the use of artificial intelligence, specifically the one used in combat situations, can increase the playing time or even the replay value of games in the action role-playing genre. Thereby, it focuses mainly on combat situations between a player and an artificial intelligence.
To begin with, this bachelor thesis examines the action role-playing genre in order to find a suitable definition for it. Accordingly, action role-playing games involve titles that send the player on a hero's journey-like adventure in which they must prove their skills in combat against virtual opponents. The greatest challenge of these real-time battles comes from the required quick reflexes, skill queries and hand-eye coordination.
Next, six means of increasing the replayability of a game are explored: Experience and Nostalgia, Variety and Randomness, Goals and Completion, Difficulty, Learning, and Social Aspect. The paper then proceeds to give an explanation for the term *Artificial Intelligence* and examines the various methods used to create intelligent behavior as well as the general advancement of the research field. Special attention is given to the implementation methods of Finite State Machines and Behavior Trees, as they are the most widely used methods for creating behavioral patterns of virtual characters.
Finally, a study conducted as part of the bachelor thesis is described, which compares a mathematically balanced artificial intelligence with a behaviorally balanced one in terms of game performance regarding the willingness of test subjects to purchase and play through the game as well as its replay value. The thesis concludes with the findings that while the behavioral approach is more promising than the mathematical approach, a combination of the two methods ultimately leads to the best outcome. Furthermore, the study shows that the use of artificial intelligence to individualize gaming experiences is promising for the future of the gaming industry.

Name: Kießling, Tina
Course of Studies: Media Informatics and Interactive Entertainment
Seminar Group: MI18w1-B
German Title: Implementierungsstrategien für Kampf-KI und deren Einsatz zur Erhöhung der Wiederspielbarkeit in Action-RPG-Spielen mit Fokus auf PVE-Kämpfen

Contents

1. Introduction

In only a few decades, the games industry has evolved from small games that are played in arcades to large productions with hundreds of developers and an average budget of several million dollars, featuring a multitude of hours of gameplay. By now, games are the biggest business sector in the entertainment industry. They are no longer a niche but have reached the popular mainstream. Certainly, such a big market comes with great competition as well. Various AAA game studios are pushing out games every year, which makes it difficult for smaller studios and developers, especially those of the indie sector, to be profitable and withstand in the market. High playtime and replayability are a major factor for keeping up with the competition, which means that "[g]ames must be designed to hold the gamers attention and draw them in for sequels and other game titles"[FGLS11, p. 5].
This holds especially true for the role-playing genre, where players usually expect to have a wide range of gameplay features as well as a high number of hours required to complete the game. With an average playtime of 48 hours to complete the currently best rated role-playing games out there [Elm19, p. 43], the pressure for indie developers is immense. Due to the fact, that indie developer teams mainly consist of only a handful of developers or even just a single person, simply adding more content for a single play through is easier said than done. Luckily, adding replayability to a game can increase its cost-to-time ratio and therefore add value and quality in the eyes of player. Even better, replayability does not require any big and expensive features such as stunning graphics, endless of different unit types, story elements or expensive voice acting. Instead, replayability requires simple, compelling and addictive challenges paired with a easy to use interface [Ada01a].

This bachelors thesis explores different techniques on how artificial intelligence can help in increasing the longevity of a game. Thereby a special focus is set on combat situations between a player and a virtual character within the role-playing games genre. Naturally, the aim of the bachelor's thesis is to explore the advancements of the research field of artificial intelligence and the common ways of how enemy behavior is created and

especially on how replay value can be achieved by utilizing these techniques.

The goal of this study is to find a method on how to increase the overall play and replay value of games with simple game mechanics. However, the purpose is not to create completely new sets of rules and ways of achieving a credible virtual character, but to explore and examine different approaches that are currently used in video game combat.

2. Fundamentals and Theoretical Background

2.1. Definition of the action Role-Playing Game Genre

Gamers around the world can already choose from a vast collection of video games with hundreds more to be released every year. For players to be able to easily select the best fit for their style of play, video games are sorted into categories based on shared gameplay aspects and characteristics called *video game genres* [Wri20].
Most genres such as *Puzzle* or *First Person Shooter* are rather self-explanatory, while others such as *Rogue-Like* demand further definition. However, finding explanations for game genres oftentimes seems to be quite a tedious act, as the definition for the term *game* itself is argued a lot, thus it is even more difficult to define the word *Role-Playing Game*, especially as there are so many different forms of role-playing games in the world and on the market, both virtual and analogue [ZD18, p. 19]. Furthermore, modern games that are labeled with the role-playing game genre are rather a mix of different game genres and stray from the classic perception of role-playing itself.

Definition of the term "role-playing"

Simply said, role-playing in the classical sense can be described as a playful way of improvisational theater. Pettersson describes it as "the art of experience, and making a roleplaying[sic] game means creating experiences" [Pet18, p. 101]. In the case of role-playing games, this experience comes with a vast focus on freedom and liberty of choice for the player. The goal is to make the player form their very own experience with their actions, as "[they] adopt a fictional identity or identities for the duration of the game. The identity may be just the player's proxy in the fiction of the game or

a more complex creation suitable for immersion and acting. [The player] act[s] in an environment that has a fictional context related to [their] fictional identity. [The player] act[s] in the game; [they are] empowered to make creative decisions" [Pet18, p. 102] which eventually leads to "the existence of quite different *styles* and *ends* of playing [role-playing game]s – focusing e.g. on storytelling, playing a role, simulating a world, or achieving goals and progress according to rules." [ZD18, 19-20].
The goal of said experience is to immerse the player into the fictional world. Players can then explore the world in all its facets through the eyes of their character, while simultaneously shaping their character's traits such as skills and abilities as they progress. Comparing it to typical literature narrative, players do not take up the role of an authorial narrator that mostly serves as an observer, but participate in the story as one of the main characters. The game world itself serves as a playground for the players in which they can make a change and have their impact on its systems as well as its history.

The beginning of role-playing as a game type

In the sense of games, this freedom and liberty in playing are best incorporated in early pen and paper and tabletop role-playing games such as the famous *Dungeons and Dragons*, which serves as the beginning of role-playing as a type of game and led to the popularity of the genre in the first place. Tabletop or pen and paper role-playing games are typically played in a larger group of four to five players or more and one game master that is guiding the party on their journey while challenging them with obstacles such as puzzles, quests or enemy encounters. The players then have to use their created character's abilities and skills to overcome those challenges [Wri22].
Unfortunately, tabletop games come with one major flaw, as preparing a session can be rather time-consuming and needs proper group management with campaigns usually being played over multiple sessions of several hours. Therefore, the act of playing comes with a lot of freedom, while the opposite is true for finding time to play. Furthermore, tabletop games are played in a turn-based manner, meaning that only one player at a time is allowed to take an action, which can significantly increase the time of one session, especially when facing a combat situation.

Computer role-playing games

With the first home computers and consoles, developers started to incorporate role-playing aspects into computer games, which led to a kind of sub-genre called *computer role-playing games*. The benefits are clear: players that own the video game and a respective device, can play it at any time for as long as they want without the need for preparing and making time for a specific session or managing time with other group members. Therefore, those games are mostly played by a single player, although plenty of multiplayer computer role-playing games exist as well.

Just like in the tabletop alternative, the player can either create their own character or they can take on the role of a predefined hero, while sometimes they even control a whole party of adventurers. Either way, the player will set out on an adventure of heroism, which oftentimes follows a simple formula where "the player starts with nothing, performs tasks for treasure and money [like] killing monsters and going on quests[...], trains [their] skills, and eventually builds [their] character into a powerhouse figure that can then right the ultimate wrongs of the land" [Sch09, p. 73]. The computer itself takes the role of the dungeon master by providing the story and challenges while simultaneously incorporating all the rules set for the game [ZD18, p. 37].

The main goal of the game is the *immersion* of the player. However, unlike in pen and paper and tabletop role-playing games, the improvisational side of the genre comes up short due to the generally scripted nature of computer games. Therefore, the story line of these games is rather linear, or at most branches linear. So even if a computer role-playing game provides different endings and ways to achieve a goal, players will usually find themselves in pre-scripted situations that at best come in different states with different character interactions and dialogues [Sch09, p. 80].
But not only the story is scripted, the players themselves are also comparatively limited in the actions they can take. In classical pen and paper and tabletop role-playing games, players could propose to do any action they want to their dungeon master, no matter if there were any explicit rules for it, who then had to decide for the success of failure of the action. Computer role-playing games on the other hand are limited to the actions determined by the game mechanics and features of the game [ZD18, p. 38]. To counteract this limitation, modern role-playing games usually provide "fully open, sprawling worlds filled with other characters, monsters, places to explore, and tons of interaction with both people and objects in the game" [Sch09, p. 70].

Interaction and game mechanics can range from questing, puzzle solving, fighting monsters and dungeoneering as well as simple actions like gathering resources and information, among others.
Generally, as stated in the beginning of this chapter, this leads to modern role-playing being more and more of a mix of different genres. The pursuing of quests may be typical for role-playing games, but the the exploration, information gathering and problem as well as puzzle solving side is rather typical for the adventure game genre, which is why these two oftentimes go hand in hand [JW01, p. 1355]
Additionally, the automatism that stems from computers taking the role of the dungeon master, led to computer role-playing games with faster gameplay. They mostly take place in real-time scenarios, meaning that players only have a limited amount of time to think and make a move, before the computer itself takes action. This change from turn-based interaction and combat to real-time speed led to more and more elements of the *action* genre that "focus on physical challenges that require hand-eye coordination and quick reflexes" [Wri20] being added to the role-playing genre and therefore creating so-called *action role-playing games*. Ultimately, these action elements led to combat and skill-based challenges being more and more of a focus in modern-day role-playing.
However, designing encounters in a role-playing game is not about creating a single opponent that is almighty but several encounters that provide challenge, but to keep the player entertained and make them feel like the true hero of the story. Encounters can either consist of small enemies that can be easily defeated, but attack in groups or bigger enemies such as boss or mini-boss fights. These "Enemies and Boss Enemies are necessary to give the player something to fight, and to provide story motivation" [Sch09, p. 90].

All these different gameplay aspects that highly focus on interaction with the world and its inhabitants, make artificial intelligence an intensive part of role-playing games. Hence, the intelligence level and focus on the behavior have to be higher than in other games. Repetitiveness or hangups in the behavior will be more noticeable, especially because role-playing games usually require large amounts of gameplay hours to complete [Sch09, p. 74].

Eventually in terms of this thesis, the expression *action role-playing games* refers to a mixture of classic role-play with the action and adventure genre. This includes games, that *set the player out on an adventure* whose primary challenge comes from a *focus on real-time combat* against *multiple different encounters* with *different amounts of*

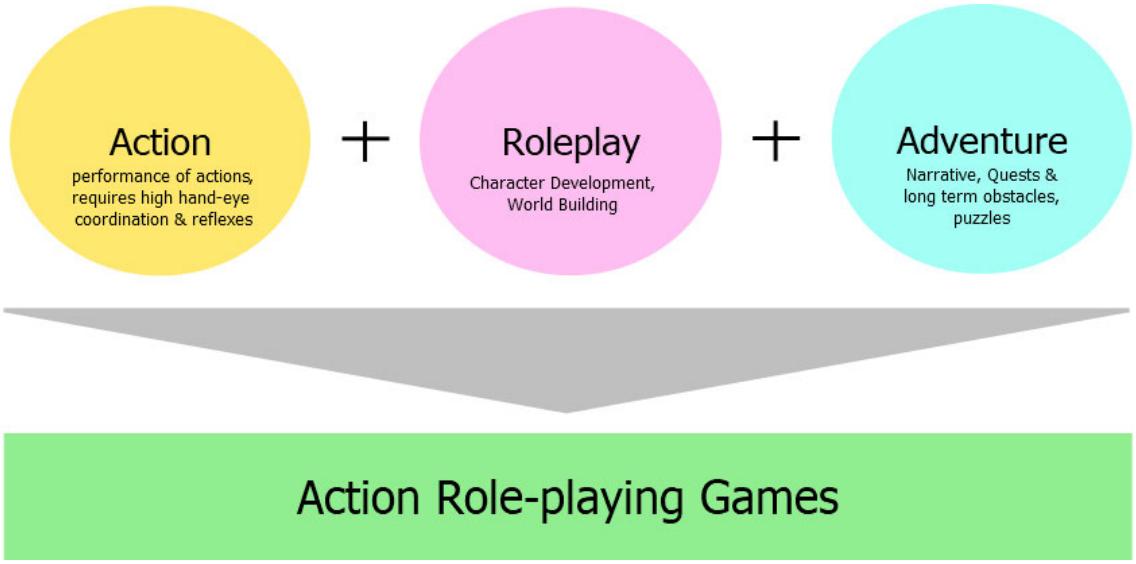


Figure 2.1.: Action role-playing games as a combination of classic role-play mixed with the action and adventure genre.

enemies and behaviors that require the player to *properly use their skills and abilities* as well as *to have quick reflexes and hand-eye coordination* to succeed and further develop their character.

2.2. Common Means to increase Playtime and Replayability

Obviously, games are not the only entertainment media available. There are plenty of books, movies or theater plays that some might consume over and over. These stories are linear and therefore do not change. Even remakes and adaptions will most likely have the same core to the story. Instead of the plot, it is their lyrical writing style, the storytelling, the world building, the performance and the emotion delivered, that make the audience want to consume these pieces again and again. Additionally, they are no documentaries and therefore do not present facts but rather stories from people from a different reality or time period but with such a high emphasize on detail, that it becomes compelling to the consumer [Ada01b].
However, the main difference between classical media and games is their interactiveness

and the option of decision-making. This opens up a completely different way of experiencing a story. However, it is important to note the joy and pleasure of playing the game is closely tied to its interactiveness. If the interaction itself ceases to be entertaining, the game will do the same [FGLS11, p. 18]. Therefore it is the designers' duty, to find a back and forth between the player and the computer, that holds the interest of the player. In order to do so, today's designers have a lot of tools to choose from, including artificial intelligence, unique game mechanics, unrealistic experiences or even physics [FGLS11, p. 18-19].

Alternative combinations of these factors will lead to different games and therefore, varying styles of play. As there are different types of players with alternate incentives to buy and play a game, it can be concluded that likewise there are as many reasons for a player to replay a game.
"Some might simply enjoy the game, its immersive story, or its satisfying gameplay and feel compelled to return for another session. Others might value customization of play higher in terms of replayability where different choices lead to different outcomes. Yet others might adopt a more focused goal-orientation as motivation for replaying a game when trying to unlock every achievement the game has to offer" [Thy14, p. 8]. Looking at possible incentives, the following six aspects of replayability for games can be formulated:

- **"The experience":** the unique feeling of a game that keeps the player coming back. This could be because of the unique gameplay, story, performance or even nostalgia. The overall feeling of the player and therefore the essence of the game is the most important factor for creating a lasting experience.
- **Social Aspect:** mostly applicable to multiplayer games, this aspect focuses on interaction among players. This could be cooperative or competitive behavior as well as simply exchanging thoughts, opinions or "the experience" with one another.
- **Completion:** the motivation to finish a game 100%. This includes but is not limited to finishing the story mode and side quests, exploring every corner of the map, levelling or collecting everything there is to offer such as hidden collectables or trophies.
- **Difficulty:** specifies on the challenge level of the game. It motivates the player to beat it in the most challenging modes and therefore drives them to their best performance.

- **Variety and Randomness:** allows the player to play a game in multiple different ways as well as offering them a unique experience with every single play-through.
- **Learning:** adding value to the life of the player by improving a certain skill and ability, whether that be in a cognitive, physical or intellectual manner.

Some of these aspects, such as difficulty and learning, may be heavily tied together but not limited to one another. Additionally, it is important to note, that all these aspects for themselves, can add to the replayability of a game, but a combination of them will ultimately hold the best results [FGLS11, p. 20-21].

To provide an in-depth view on what these aspects mean and how they can be incorporated into games, some of them will be explained in more detail in the following sections.

"The experience" and nostalgia

"The experience" is probably the toughest of the six presented aspects to grasp. It has a lot to do with the feelings, that games convey and most importantly invoke in the player. In terms of traditional media, there are a multitude of factors that need to harmonize with each other to create a coherent image and achieve the desired effect. In the case of theater and musicals, that is the stage design, the performance of the main and supporting actors, the selected songs, dance numbers and, last but not least, the spoken word. They all play an important part in conveying a certain feeling and immersing the audience in the world of the play so that they will forget the world around them for the duration of it. If the feeling is captured right, it does not even matter how often a single person has seen the play already, they will still be able to find joy in it [Ada01b].

With games being interactive, designers have even more options to create a certain feeling and experience for the player. "All it requires is knowledge of that experience and how to recreate it. In [the case of bungee jumping], that means a large elastic rope. In more complex cases, that can mean the structure of a game that partially defines how the game is played" [FGLS11, p. 13]. This structure will mostly be accompanied by a formal set of rules.
Rules can define the artificial world of the game to create a clear situation while simultaneously restricting the amount of possible actions of the player. While this contradicts

with the "ideals of freedom" that would be typical for role-playing games, it enables the designers to replicate a specific scenario and therefore an experience and its associated "magic circle"[1] [SZ04, p. 4-6].

"Without rules, the activity expands to a ludic activity. This causes the activity to only reproduce a general feeling of risk, competition, disorientation or surrealism instead of a specific experience within one of those domains. Without structure, the activity becomes purely playful and simply gives the essence of freedom. [...] Every time this combination of structure and rules is imposed, it cuts out a very specific experience" [FGLS11, p. 13-14].

While it is important, that a situation can be replicated, only allowing this single way of playing a game, e.g. by incorporating too many rules, will ultimately lead to less fun. Games with only a few variables will be more limited and predictable. This can lead to the player being in the same situation and facing the same behavior over an over. Eventually, the overall challenge decreases due to the fact that the player can predict what will happen by e.g. *reading* the opponent. This is a state that has to be counteracted, as it makes the game prone to quickly become boring [BCCT20, p. 330]. In order to circumvent this problem, it is important to integrate slight unpredictability and uncertainty such as "math problems [the player does not] know the solutions to, or more variables (and less predictable ones) such as human psychology, physics, and so on" [Kos13, p. 38]. Therefore, this form of creating an experience is directly linked to the aspects of difficulty and variety.

The importance of creating an experience is most evident, when having a look at games that are mostly played for *nostalgic* reasons. Certain games within e.g. the *Metroid*, *Zelda* or *Final Fantasy* franchise, were created more than 20 years ago, but are still loved and frequently played. Therefore, a lot of game companies started to remake or remaster old game titles. This will not only bring the games into a new light, it also

[1] The "magic circle" is basically a space and frame in which a game takes place, that separates the world of the game from the outside world. It is "a concept connected to the question of the 'reality' of the game, of the relationship between the artificial world of the game and the 'real life' context that it intersects" [SZ04, p. 2]. The "magic circle" also serves as the frame for the experience. Thus, breaking it will ultimately lead to a "break" in the experience, which is why the rules and mechanics of the game have to be adjusted to serve the needs and the experience of it. A fighting game for instance should not force the player to click through menus, inventories and notes more than it does provide combat

enables designers to refurbish the experience on new technology and console generations [FGLS11, p. 30-31].

Goals, completion and social aspect

"A game's goal is often the largest single element that drives the pleasure of a player. The goal is the ostensible reason for playing, but the goal is never easily attained; rather, it is the obscure object of desire, the carrot held just out of reach, pulling players forward through the varied pleasures of game play" [SZ04, p. 14-15]. Therefore, a game needs a goal. However, this goal may not be the *ultimate goal* and therefore the end of the game all the time but can come in the form of several smaller goals along the way. Role-playing games are typically constructed in that manner.
Additionally, the goal may not always lead to the same ending. For instance, some games incorporate multiple different endings, depending on the difficulty setting, decisions that were made while playing or other conditions. While the ultimate desire to complete the game remains the same, it encourages players to achieve the best ending possible [FGLS11, p. 17].

Besides the obvious goals made by the game and its story, there are multiple ways of generating additional incentives for the player to play in a different way and therefore, play the game again. This could be finding all Easter eggs, collecting or crafting all items or other collectables within the game as well as achievements and unlockables. "Achievements and unlockables ask the player to accomplish difficult or unusual tasks within the game[...]. If a player is aware of such achievements or unlockables, they may replay the game to see if they can get it the second time around. In addition, these often ask you to play the game in a less obvious way" [FGLS11, 24-25]. Adding in any kind of statistic or collectables to a game will inevitable lead to some players aiming for the 100% run.

Sometimes its even the players themselves, that make up goals and challenges in order to beat a game. Events like "Games Done Quick" feature multiple speedruns of a variety of games and is even used to collect money for charity. On the other hand, there are certain challenges that place an additional set of rules upon the player such as the "Nuzlocke"-

challenge[2] for the Pokémon games or the "Zero Death"-Challenge[3] for Dark Souls.

Challenges like that are oftentimes created by community members themselves. Naturally, this leads to a lot of exchange regarding the game and the challenge as well as countless social gatherings like the previously mentioned events.
In general, games nowadays "bring a new level of statistical information to players, giving them an uncanny level of information to rank their skills and compare with others. Using this statistical information allows players to judge their progress and compare their achievements with others all around the world. This massive collection and display of statistical information creates a sense of camaraderie, rivalry and accomplishment. All of which furthers the desire to continue playing a game" [FGLS11, p. 20].
But the social aspect of games is of course not limited to the discourse about a game or the player's accomplishment. Instead, a lot of games incorporate multiplayer features which can either result in the players competing or working together while playing. Especially the former leads to a great sense of replay value as can be seen with various Player vs. Player games such as *League of Legends*, *Call of Duty* or *Overwatch*. "Human opponents are more interesting [than virtual ones] because in addition to having varying strategic and tactical abilities, they differ in the degree to which they're aggressive or defensive, devious or forthright, cautious or risk-takers. And of course, you can talk to them. There's a social aspect of playing against other people that is completely absent when playing against a computer, and that tends to make the game replayable even if nothing else does" [Ada01a].

Difficulty

It is unavoidable that the fun of a game is bound to its difficulty. So if a game fails to provide a challenge, it becomes boring. However when looking at difficulty as a means for replayability, it has a lot to do with which type of gamer a player is. When it comes to general game-playing, there are two different types of players: the casual gamer and the core gamer. The casual gamer plays for the joy of playing and seeks for variety

[2] A challenge that adds more rules to the game such as that the player is only allowed to catch the first Pokémon they encounter on a new route or a Pokémon that drops to zero health points has to be freed because it actually "died".

[3] As the title suggests this means, that the game is supposed to be played without the player dying within the game. Further, the challenge level can be increased by not being allowed to be hit at all or attempting the run with a specific character built.

in doing so (see the section about Variety and Randomness in this chapter for further information), on the other hand, the core gamer's primary motivation is beating the game no matter the challenge. Their motivation is kept high, as long as the gameplay is interesting and challenging. While this criteria is met, the core gamer will be happy to repeat a game or a passage of a game over and over, until they succeed. However, as soon as they beat the game or tire of the gameplay and get the feeling that they simply cannot improve anymore, they will eventually stop playing because there seizes to be a challenge they can overcome [Ada01a].

Usually, games will eventually have an increase in their difficulty, as the player progresses. This could be incorporated by a difficulty curve that increases the value as soon as the player progresses over a specific part of the story or had the opportunity to obtain a certain item or ability [Byc09], but also by simply providing a range of difficulty settings for the player to choose at the beginning of the game. The latter enables core gamers to increase the overall challenge if e.g. the "normal" setting becomes to easy. Casual gamers on the other hand could usually play on a "easy" or "story" mode.

Balancing out the difficulty of a game has a lot to do with understanding the theory of flow by Csikszentmihalyi. In general, flow is a total state of involvement and *being in flow* means being in a "state in which action follows upon action according to an internal logic which seems to need no conscious intervention on [the player's] part" [Csi14, p. 136-137].
It is a state that one enters, when the demands of a game align with the own skills. While in flow, the player feels that the right amount of call for action based on their skill level is put onto them. As shown in figure 2.2, if the demand is too high, they will feel anxious and overwhelmed. On the other hand, if the demand is too low, they will quickly become bored and feel underwhelmed [Csi14, p. 146-147].

In order to provide some sort of *fail-save* for when the state of flow is disrupted by an obstacle that seems *too hard*, games often provide the player with a chance of training or leveling up for a specific encounter. That means, that "[w]hen a player encounters a portion of the game that is too difficult, they must replay it or practice in another portion of the game until they are able to pass it. [...] Various games employ this strategy to keep a fairly constant risk of failure. The better a player does in these games, the harder it gets and the more likely they will fail and therefore, replay" [FGLS11, p. 22].

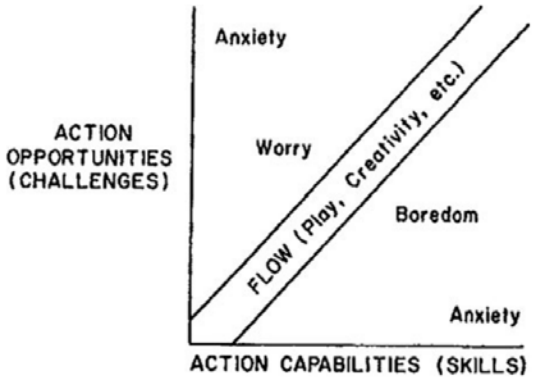


Figure 2.2.: Flow activity chart according to Csikszentmihalyi, that shows the relation of the challenge of the game and the skill level of the player and the invoked emotions and feelings [Csi14, p. 147].

Nonetheless, games should still occasionally alternate action with rest phases in a level, even while overall increasing the difficulty with progress. The latter primarily ensures that players do not burn out as quickly as if they were permanently exposed to unpredictability or stress due to encounters with opponents and other obstacles [Sch09, p. 37-38]. Besides, having to face challenges that are only beatable with a lot of practice will be frustrating for some players, leading them to stop playing and not pick up the game anymore [FGLS11, p. 22-23].

Learning

The aspect of learning as a means of repeatability is heavily tied to the difficulty of a game. Likewise, it is closely related to behavioral theory. Almost everything humans do and like to do is based on the basic aspect of learning or improving a skill or knowledge they have already learnt. In this case however, learning is not tied to going to school or university, doing homework or studying for an exam. Instead, it focuses on the activities a person enjoys doing and considers their hobbies. Learning may come in many forms:

- **Sports:** exercise the body and the muscles, learning how to move in a certain way. Especially demanding sports even exercise the brain at the same time.
- **Reading books:** most obviously, it will train the reading and writing skills. The brain gets to know new words. Additionally, depending on the subject of the book,

books can teach a person about certain aspects of science, the society, the *"old days"* etc.

- **Talking to people:** it does not really matter weather the person talking to is a friend, family or a complete stranger, but by actively listening to others, a person can easily broaden their horizons and maybe even see aspects of their life or surroundings in a new light.
- **Travel:** getting to know and experiencing the culture of a different country. Simply seeing other places and enjoying the locations, the food, the culture, the people and their traditions.

The list goes on.
Just like these examples, games only remain interesting for a longer time, if they add value to the life of the player by exercising their brain or other parts of the body. Just like in everyday life, if one always does the same things over and over again, the activity will eventually become second nature which will result in the brain not being engaged in the activity anymore. As soon as that happens, boredom is inevitable [Kos13, p. 38-42].

That is why the best games follow the rule of "easy to learn, hard to master". This means, that "good games [should be] able to constantly teach [the player] more about how to play them" [YT18, p. 10]. Thinking of games like Tic-Tac-Toe, there is not much of a challenge as there is only a limited amount of moves that are possible and just after a few turns, it already becomes clear how the game will end. Chess on the other hand drastically raises the number of possible moves. The learning aspect of games is inevitably the reason, why they are such a popular test-bed for artificial intelligence implementations as well [YT18, p. 93].

Depending on the genre of the game, a multitude of learnings can be drawn from them. Strategy and board games may train the strategical and analytic thinking side of the brain, while puzzle and adventure games can exercise problem solving skills. Likewise, action games enable the player to train their reflexes and hand-eye coordination. Learning is not always about getting it right on the first try, but to keep practicing until a skill is mastered.

One of the best genres to incorporate this style of design is the souls-like genre. It is a sub-genre of the classic role-playing game, that originates from From Software's

"Demon's Souls" and "Dark Souls" franchise. Their formula of how to create combat and challenge has become widely popular among players and designers because it conquers the art "of learning through practice [and] mastering the combat, [which will inevitably lead to a great] sense of achievement when [the player] finally beat[s] an enemy [they have] been struggling with for a while" [Bai21].

Variety and Randomization

Variety and randomness add novelty and excitement to the game. Role-playing games are especially known for providing a vast variety of content to the player. In the role-playing genre, this can come in the form of different classes or races, character development, game mechanics, a vast number of individual quests, virtual character personalities, enemy designs and encounters, cities, dungeons etc. Lately, a lot of game titles are even advertised with bringing variety to the story by giving weight to the decisions of the players.

Great variety in games is especially important for targeting the casual gamer. They do not necessarily need challenge in order to continue playing a game but enjoy a great sense of variety and content [Ada01a].
Creating all this amount of gameplay can take up a lot of manpower and especially a lot of time. Thankfully, new technologies such as procedural content generation can assist developers in creating more variety of content nonetheless. It helps to automatically let the computer generate non pre-scripted worlds and levels or even dialogues and quests automatically at runtime, which in return provides the player with far more content that - if implemented well enough to create sufficient variety, quality and quantity - can create open-ended games as well as add great replay value [ZD18, p. 38][YT18, p. 24].

But in order to create variety, developers are not necessarily required to add more and more content. Instead it can suffice to simply provide the player with different ways of playing a game or adding in *random* elements that add a sense of uncertainty so that even if the initial state of the game is the same, it can unfold into a different direction. A good example for this would be classical board games where the initial state of the game remains the same but the course of the game is determined by throwing dice or games that provide the player with varying starting conditions such as a different starting hand in card games.

Of course it is not only possible to create different play sessions by using random aspects. Instead, additional features such as classes or a skill tree enable the players themselves to create different experiences with every play-through, as they can shape their character differently each time. This will most likely result in adapting to a different style of play for the player in order to finish the game. That is why it is also important to create the possibility to deal with obstacles such as puzzles and encounters in multiple different ways, even if the ultimate goal and outcome might be the same [Ada01a].
Besides, "it is crucial in a game that players don't know exactly how it will play out. Think about it: if you knew who was going to win a game before it started, would you even bother to play" [SZ04, p. 174]?

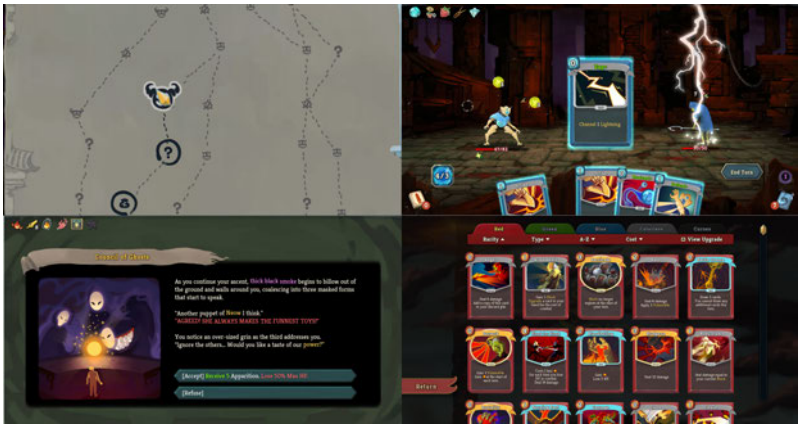


Figure 2.3.: *Slay the Spire*: a rogue like card game, available on Steam, Switch, Playstation and Xbox.

One very good example for this is the rogue-like card game *Slay the Spire* (see figure 2.3). Just like any other rogue like, the player is tasked to complete procedurally generated dungeons with multiple stages that include encounters and mini-bosses as well as a final boss. The structure of a dungeon has a similar yet alternate structure each run and consists of several rooms, that can either be enemy or mini-boss encounters, a campfire at which the player can restore health or improve their deck, treasures, shops or rooms that provide a short story followed by a decision. While progressing through the dungeon, the player can improve the strength of their cards or add new ones to their deck as well as find special relics that provide passive bonuses throughout the fight. Additionally, the player can choose between four characters which come with different deck builds and styles of play such as a typical Barbarian or Rogue. The randomized structure of the

dungeon paired with the deck building aspect creates a huge amount of possibilities for styles of play each run, which keeps the game interesting.

Furthermore, it is possible to create incentives for players to replay a game by simply adding in new content such as events or other downloadable contents that e.g. provide a new story arc. However, in order for this strategy to work, timing is crucial as players may have already moved on to a different game, while adding it too soon will result in players seeing new content as a given part of the game, which can result in players abandoning the game sooner, especially when the developers do not keep on providing more [FGLS11, p. 31].

Critical look at replay value

As declared previously, adding replay value can drastically increase the quality and value of a game in the eyes of the player, as they easily get the feeling of *"getting their moneys worth"*. However, in order for a games company to make profit, it is necessary to balance out the amount of replayability with the need of the player to simply get more. On one hand, replayability is necessary to grow popularity for a franchise or future games of the development studio, on the other hand, players need to have an incentive to purchase other games as well as future productions [Ada01a].

2.3. Artificial Intelligence and its Relationship with Video Games

In modern-day society, everybody, even from a young age, has heard of the term *artificial intelligence* at least once in their lives. Even more so, everyone owning at least a smartphone comes in contact with artificial intelligence every day. It can be found everywhere, whether it is in filters on Social Media, Smart Home Devices, Navigation Technology, search algorithms, recommendation services of streaming platforms of any kind, dating apps or even things as simple as the auto-correct function on phones or computers. Artificial intelligence makes the lives of humans easier by helping them do and find the things that matter to them more easily, while also keeping them entertained. Consequently, it is not astonishing that a large variety of aspects of artificial intelligence

are used in one of the largest entertainment sectors today: video and computer games.

However, although artificial intelligence is such a profound aspect of everyone's lives, the term itself remains unclear for the majority. The following chapter serves as a means for clarifying the concept itself. Furthermore, it will highlight the relationship and bond between video games and artificial intelligence and how it progressed from its beginning to its current state.

2.3.1. Basic Understanding of Intelligence and Artificial Intelligence

Definition of Intelligence

In order to be able to understand the term *"artificial intelligence"* it is first important, to define and understand the term *"intelligence"* first. Finding a definition for the term is rather complicated, and has been debated for a long time. Generally speaking, intelligence is "the capacity to acquire and apply knowledge [that] demonstrates the faculty of thought and reason" [Sch09, p. 2].
However, this definition can be further expanded. In fact, many different forms of intelligence exist that address different parts of the brain. These forms include but are not limited to linguistic intelligence, critical thinking, abstraction, logic and problem solving, spatial intelligence, creativity, body coordination, planning, self-awareness as well as empathy or awareness towards others and towards the natural world. Each of these forms of intelligence work as a separate system, although they can interact with each other to create intelligent performances [Ste15, p. 229]. The distinctness of each intelligence varies from person to person, but can be further enhanced and manifested through training and practice [Sch09, p. 11]. *"True intelligence"* is demonstrated, when one can acquire knowledge and apply it within a variety of contexts.

Definition of Artificial Intelligence

Artificial intelligence on the other hand is the modelling process of intelligence with especial focus on human intelligence. Its "aim [is] to make computers perform tasks that have at some point been considered to require intelligence to perform" [YT18, p. 7].

These tasks can require any of the previously mentioned forms of intelligence and can be either of cognitive or behavioral nature and depict human and rational thinking, though the two sometimes contradict [Sch09, p. 2].
"[I]n practice, the term refers to a spectrum of ideas ranging from a simple system that can perform only basic tasks to a fully adaptive system that is able to solve highly complex problems by using techniques that reflect the nature of human intelligence" [JW01, p. 1356]. Depending on the underlying research background the importance of modeling true human cognition varies. Naturally, the research field has "two main goals. First is [...] understand[ing] intelligent entities, which will, in turn, help [...] understand [humans]. Second is to build intelligent entities, for fun and profit, [...] because it turns out that these intelligent entities can be useful in [...] everyday live[of humans]" [Sch09, p. 8].

In case of this thesis, the focus lies on games and the behavior of the virtual opponents within the game. Therefore, it is not necessary to create human cognition or even simulate learning but simply to simulate behavior that *seems* intelligent.

The relationship between Artificial Intelligence and Games

For decades, video game development and research into artificial intelligence modeling methods have influenced each other. Insights and breakthroughs in one of the two areas always ensure progress in the other [BKT14, p 537]. This is not at all surprising "as games were one of the first software systems interacting with humans on a rather high level" [Kuf18].
Furthermore, games themselves require effort and skill to be completed which are tied to cognitive and behavioral intelligence. Games usually provide the player with a variety of possible moves and actions to choose from, which leads to a variety of strategies for winning and finishing the game. These strategies are modeled by using a tree structure, with the nodes of the tree being the different states the game can end up in and the edges being the action or move that leads to that state. Due to the vast number of possibilities of actions and states that can even exceed the number of visible atoms in the universe, these trees naturally grow big quite easily. Hence, calculating the *best* move to win becomes a much harder and more complex problem for computers than for humans, which overall is the reason why games are a particularly exciting subject for computer scientists and thus artificial intelligence research [YT18, p. 15-16][Kuf18].

When looking at the different problems games provide for artificial intelligence implementations and methods, they can be sorted into four different subcategories:

- **perfect information games:** games in which all players are aware and informed about the moves and actions of the other players as well as the current state of the game and all possible moves and actions. Classical examples are two-player board games such as *Chess* or *Checkers*.
- **imperfect information games:** games, the current state of which is not fully known to all players. Typically card games with face-down cards such as *Poker* fall into this category.
- **stochastic games:** games whose possible moves and actions are determined randomly and can change each turn. This includes all games that are played using dice or another form of randomness such as *Backgammon* or *Ludo*.
- **interactive games:** games that provide a repertoire of actions and different strategies for the players to pursue in order to be able to reach a certain goal. Sometimes they even provide the player with the possibility to define their own goals. Usually, video games as well as classical pen and paper games fall into this category.

Obviously, video games are overall categorized as *interactive games*. However, parts of a game can also fall into one of the other sub-categories, making it a mixture of different problems to solve. This is especially true when games feature mini-games [BKT14, p. 532-534].

Generally, the foundation of video games is the same: they consist of a staged scenario created by the game designers and developers. The scenario on the other hand, can be fairly different. Some video game genres such as real-time-strategy games are fairly similar to classical board games and therefore serve as an *extension* of them, while others such as the role-playing genre are heavily influenced by movie culture. Therefore, they are more like interactive movies and stories. Depending on the amount of scripting and streamlining, the player can either create their own story in a rich environment or follow a scripted and streamlined path. The latter usually follows a rather scripted approach when it comes to the implementation of behaviors, that follow the designers intentions and that, due to the scripted nature, contradicts with the usual understanding of artificial

intelligence [BCCT20, p. 328-330].
In general, agents for interactive games require a mixture of strategic and reactive behaviors which is not at all trivial to achieve, especially with the action space oftentimes being much bigger than the one of classical board games [BKT14, p. 535].

Requirements for Non-Player behavior in interactive environments

One of the key topics for artificial intelligence in video games is the development of virtual characters and their behaviors. In order to be considered *good*, they have to be credible, somewhat realistic and imitate intelligent behavior. Oftentimes, they are the reason for players to feel engaged in a game as they provide the source of conflict or story. Virtual characters can either take the role of an opponent or an ally. The first being the one that is most key to this thesis. Additionally, the artificial intelligence of a game has become one of the major promotional materials in the marketing of games. Therefore, its implementation and design is capable of either making or breaking a game [BCCT20, p. 329][Kuf18].

In case of games, it is the behavioral mechanics of the game that is oftentimes referred to as artificial intelligence, though the term is not particularly true and is used for close to everything that is steered by behavior of a character or the environment in the virtual world. However, in fact the mechanics used are oftentimes the same functions and mechanics a player is using and the algorithms and structures triggering them are mostly scripted [Vya20][Sch09, p. 3]. This is especially the case, because most game genres that are known today have evolved from earlier video game designs where artificial intelligence was simply not as present as today (see chapter 2.3.2 for further details) and have been designed to not need it in the first place [YT18, p. 14].
For the player and the designer of games, it is only really important to know the outcome of behavior, not how it is achieved or how the algorithms work. Designers intent to create an experience that players can enjoy [Sch09, p. 7]. Therefore, unexpected behavior is not desired, which is why true learning algorithms are rarely used in most video games. Instead, it is oftentimes behavior trees and state machines coupled with fuzzy logic, that are doing the trick. The current state of the character is abstracted using characteristic elements such as its position in the world, available tools and weapons as well as the distance to the player and their temper among others. Depending on the values, the

virtual character then forms decisions on what to do such as where to go or which action to take. Depending on the game genre, those strategies can be further mixed with other methods on how to create artificial intelligence [BKT14, p. 534-535].

Depending on what designers hope to achieve from the creation of artificial intelligence, other criteria arise that must be met. If there is a desire to create a certain experience, predictable and often rational opponents are used, whereas characters that are supposed to simulate a human opponent must have a spark of unpredictability, just as human behavior is sometimes unpredictable and not rational [YT18, p. 96]. However, the latter is oftentimes not as fun for the player [Vya20]. This fact, that artificial intelligence and its behavior is created in order to fulfill design standards makes balancing and perfecting the implementation so hard. In order to make the players happy and give them a good experience, the designers are "better off having an [artificial intelligence] that was just above average all the time, rather than one that was brilliant 98[%] of the time and stupid 2[%] of the time" [Cas02, p. 41]. This can only be achieved by continual testing and iteration. So at one point the creation of behavior becomes some sort of trial and error which results in it "not [being] the best code[, but instead] it is the best use of code and a large dollop of 'whatever works'. Some of the smartest-looking games have used very questionable methods to achieve their solutions [in order] to give the illusion of intelligence and enhances the fun factor of the game" [Sch09, p. 7].

When looking at the behavior of characters in games, it is also important to note that the players themselves do not necessarily play for the sole purpose of winning, beating or completing a game. Instead, they may play because they want to pass time, relax, explore etc. Creating an artificial intelligence that simply wants to win can sometimes even lead to behavior that is not at all predictable or fun to watch. The opposite might be true. So the focus should really lie on how to create a *human-like* and *believable* behavior, instead of one behavior that is superior and beats the player every time [YT18, p. 91][Sch09, p. 8-9].
This *human-like* and *believable* behavior means, "that [the virtual character should] ha[ve] performance comparable to a human, has similar reaction speed, makes the same sort of mistakes that a human would do, is curious about and explores the same areas as a human would" [YT18, p. 94] and so on.
Enhanced reflexes and skills on the other hand would not only create incredible non-human-like behavior but could also discourage the player from playing, as they may get the feeling that it takes otherworldly reflexes in order to be able to use the mechanics and

Table 2.1.: The two goals an artificial intelligence implementation can aim for (win or experience) and the two roles it can take (player or non-player) [YT18, p. 92]

	Player	Non-Player
Win	**Motivation** Games as AI testbeds, AI that challenges players, Simulation-based testing **Examples** Board Games AI (TD-Gammon, Chinook, Deep Blue, AlphaGo, Liberatus), Jeopardy! (Watson), StarCraft	**Motivation** Playing roles that humans would not (want to) play, Game balancing **Examples** Rubber banding
Experience	**Motivation** Stimulation-based testing, Game demonstrations **Examples** Game Turing Tests (2kBot Prize/Mario	**Motivation** Believable and human-like agents **Examples** AI that acts as an adversary, provides assistance, is emotively expressive, tells a story, ...

beat the game. This is also applicable for non-human looking creatures and opponents. "The goal [for creating virtual characters and their behavior] is not to reach human level [intelligence] anymore, it is to propose an opponent, or a game companion, with whom the human player will enjoy the confrontation. This touches on some complex issues related to the notion of enjoyment and entertainment, at the heart of gaming" [BCCT20, 332].

Creating these different confrontations, whether they are of friendly or hostile nature, also changes the requirements for the behavior and its design itself as can be seen in table 2.1. Additionally, artificial intelligence can even be used to control and steer the

players themselves, e.g. when switching to an autopilot-mode, for gameplay showcasing, simulating human opponents or simply for easy play-testing purposes.

2.3.2. A brief History of Artificial Intelligence in Video Games

In the past 30 years, interactive computer and video games have become one of the most important research fields and areas of application for artificial intelligence. Nowadays there is close to no modern video game that does not make usage of the latest artificial intelligence models and their algorithms [BKT14, p. 534]. "Thus, [it is no surprise, that] the use of [artificial intelligence] techniques has increasingly become a necessity in order for a game to stand out in the market. Moreover, as more games incorporate functional and impressive [artificial intelligence] techniques, users' expectations and demands increase and a game that fails to employ effective [artificial intelligence] is less likely to be well received" [JW01, p. 1356]. But obviously this is the current state of the industry and was not always the case in the past. In fact, research on artificial intelligence as a general topic in academics dates back as far as the 1950s, making the subject itself a particularly young one in the sense of science [Sch09, p. 1].

The history of artificial intelligence with games differs when looking at analogous games, such as board or card games, and computer or video games. As stated in chapter 2.3.1, in terms of artificial intelligence, games can further be sorted into four different categories *"perfect information games"*, *"imperfect information games"*, *"stochastic games"* and *"interactive games"*. Naturally, just like with genres, more than one may apply to a game at a time. Typically most analogous games fall into at least one or more of the first three categories, while computer and video games naturally classify as *"interactive"*, but can also incorporate parts of the other three categories.

Artficial intelligence for board games

Early research on artificial intelligence in games initially focused on games that were classified as *"perfect information games"* with particular focus being placed on the games of Chess, Checkers and Go. Just like most other board games at the time, the two most required skills for a player to be able to play and perform well were analysis and reasoning [BCCT20, p. 327]. Typically these games consisted of a staged conflict between two

players. Games with these specifications were particularly easy to formulate in terms of artificial intelligence research because the moves executed lead to only a limited number of possible game states that were clear, comprehensible and predictable to all parties involved. These states can be easily modeled using discrete structures, the most commonly used one being search trees.

First approaches to programming a chess computer, by means of search trees, were already proposed in the 1950s by Claude E. Shannon. Shannon's proposal was to calculate every possible state the game could end up in within a limited amount of moves starting from its current state. These states and the moves that lead to them were then stored in a search tree. Cost functions would be used to measure the goodness of the respective move, with particularly good ones being the ones that either maximized the own or minimized the opponent's chance of winning. However, first implementations of this simple tree search method quickly proved to be impractical due to the exponential growth of the size of the tree and the resulting computational cost. Hence, the approach was later further improved by e.g. using heuristics that allowed the algorithms to only expand on the nodes that proved to be especially promising, resulting in some of the most known and still well used approaches such as the alpha-beta pruning or the A$*$ algorithms [BKT14, p. 532-533].

The fact that these and similar algorithms, as well as the application of machine learning techniques, are still widely used today proves, that solving problems related to *"perfect information games"* is a major achievement in the history of artificial intelligence. Thanks to major breakthroughs such as the DeepBlue Chess bot or the Logistello Othello bot, both of which beat their respective world champions as early as 1997, these types of games and related problems can be considered solved within the research field [BKT14, p. 532-533]. Even the world champion in the game Go, which until then was still considered too difficult a problem, was defeated by AlphaGo back in 2017 [YT18, p. 16].

Artficial intelligence for computer and video games

Turning the focus on computer and video games, history is rather different. One of the reasons for that is the fact that they fall into a different category of artificial intelligence problems than the previously examined board games. Most obviously, they differ in being

"interactive games". Usually these games have a larger set of actions to choose from, but the information about the moves, stats and abilities one character holds are not necessarily openly communicated to the counterpart. This drives video games more into the direction of *"imperfect information games"*. Consequently, that means that unknowns and probabilities have to be taken into account when predicting and calculating possible moves, which in return complicates the classification of *good* and *bad* ones. In result, classical tree search was not efficient enough to solve the problems that come with the creation of behavior of virtual characters in computer and video games.

The complexity of the problem lead to artificial intelligence being not being prominent at all in the first video games released in the 1970s. This was mainly due to the early state of research and general hardware limitations such as the shortage of processor speed and the lack of memory storage [YT18, p. 11]. Consequently, even though virtual characters and their behavior was a large focus for development even in early days, the first implementations of behaviors of Non-Playable Characters were very simple and mostly scripted [YT18, p. 11] meaning they mainly consisted of repetitive and predetermined patterns of a certain type of motion and attack which were chained together in a seemingly *"random"* fashion[4].
The main reason for the usage of this method was its easy and efficient way to cope with said hardware limitations due to the fact that "[p]atterns could be stored easily, requiring minimal code to drive them, and required no calculation" [Sch09, p. 5]. Accordingly, the key difference between classical board games and most video games was their way of providing challenge to players. Board games mostly demanded situational analysis and reasoning, while their digital counterpart was highly interactive and therefore heavily required quick reflexes paired with pattern analysis [BCCT20, p. 327].

In order to still be able to provide a challenge for the player when the implementation fails and the opponent therefore falls behind in competitive situations, early behavior development oftentimes involved cheating mechanisms for the virtual characters. This meant, the artificial intelligence had additional information about the game and the world or certain skills that the player did not have and was not able to acquire. This could be accessing certain variables that determined the state of the game or the player and use

[4]Since computers are generally deterministic systems that allow little unpredictability, in the context of computer programs such as video games, there is no such thing as true *"randomness"*, which is why computer scientists refer to it as so-called *"pseudo-randomness"*, which is based on special mathematical algorithms and calculations.

that to their advantage or manipulating random factors or physics.
An example for the first could be tracking down when a player is pressing the attack key in order to be able to perform a block or dodge in the right moment or knowing where to access certain resources in a strategy game without the need of searching for it. The most prominent example for the latter would be rubber-banding in racing games, meaning that e.g. virtual characters that fall behind get better items or simply drive faster than what is usually possible in order to catch up with the player and thereby rejoining the competition.
These cheating mechanisms are meant to create a more challenging situation for the player, however they had to be subtle enough to slip past their attention. Otherwise, they create the opposite situation and therefore lead to less satisfying opponents that leave the player feeling betrayed rather than entertained and immersed [BKT14, p. 534][Sch09, p. 5].

Evidently, considering the simplicity of the behavior of most virtual characters and systems in video games' early days, artificial intelligence was not their primary selling point. Instead, game development focused heavily on making games look good by ultimately increasing their overall graphical look. This "early emphasis on graphics eventually led to specialized graphics processors on almost every platform, and the main CPU[, whose specification has likewise greatly improved over the years, was] increasingly being left open for more and more sophisticated [artificial intelligence] routines" [Sch09, p. 6], meaning that instead of devoting 1 to 2% of CPU time to calculating artificial intelligence elements, today's standard lies around 10 to 35% or even higher than that [Sch09, p. 6] while at the same time with the increase of "the average home PC['s specification], the same percentage of CPU processing power allows far more advanced techniques than it did three years ago[5]" [JW01, p. 1356].

With these advances of hardware capabilities, scripted behavior that relied on patterns such as moving from left to right and occasionally shooting was no longer sufficient. Instead, especially for the genre of adventure and role-playing games, the focus shifted towards a more sophisticated behavior that would make virtual characters pass as *intelligent*, with the main goal of making them more *realistic*, *credible* and ultimately more *entertaining* and *immersive* for the player [BCCT20, p. 327]. That means, that the virtual characters not necessarily *have to be* intelligent in the classical goal of artificial intelligence research, but rather need to *seem* like they are behaving intelligently. So

[5]published in 2001, therefore referring to the late 90s

the goal really was to *mimic* and *replicate* a behavior that simulates the same cognitive skills and abilities of a human player.

This of course makes for a not at all trivial task to achieve, especially because true human like behavior requires a rather elaborate combination of re-activeness paired with tactical and strategic thinking. This means, that virtual characters need to be able to perform quick actions in regards to the current state of the game while also planning ahead in regards to medium-term as well as long-term goals. This intricacy emerges from the fact that games themselves become more realistic and therefore complex. In contrast, a ghost in the early arcade game Pac-Man did have a lot less tactical options and behavioral capabilities than a teammate or opponent in a modern day action game or shooter [BKT14, p. 535].

Modern day artificial intelligence in the games industry

Modern day artificial intelligence is oftentimes still created by using rule based systems such as behavior trees but combined with fuzzy logic. This means, that the current state of the character is characterized using a variety of abstract variables such as their position, available items, distance to other players, etc. Depending on the state, reasonable actions will then be generated by either using deterministic or random methods. This evaluation allows for rather elaborate behaviors as well as team play mechanics.
Over the years these techniques have been further improved and adapted to different games and their needs [BKT14, p. 534-535]. This also leads to the usage of more complex artificial intelligence techniques like heuristic search, learning and planning. As a result, by the standards of academic research, modern artificial intelligence in games is evolving more and more in the direction of *real* intelligence, so that predefined and scripted patterns and behaviors are eventually becoming a matter of the past [Sch09, p. 5]

Still, classical learning algorithms take a significantly longer time to develop and design than is the case with classical behavior pattern creation. Learning usually requires many iterations, balancing as well as readjusting by designers and developers in order to work the way it is intended to work. In a game with significantly more opponents, this could drastically increase development time, which is why it is necessary to balance the pros and cons of machine learning before incorporating it into the project.

This may change in the future when speed and accuracy are further increased, but for now, most games that use learning technique only do activate it during production and then ship the game with the learning disabled to keep the behavior as stable as possible [Sch09, p. 14].

Nowadays, some problems of artificial intelligence in video games can be considered solved. This mainly applies to simple puzzle games or games that require dexterity and skill. Jump-and-run games such as Super Mario whose main problem is the navigation through the level can easily be solved by using pathfinding algorithms such as A$*$.
However, despite major advancements in terms of other problems considering behavior, there is still no ultimate and generic solution that can be applied to any type of game and genre in sight [BKT14, p. 535]. "Right now, game [artificial intelligence] is still very game-specific and very much in the hands of the coders who work on it" [Sch09, p. 7-8].

Still, "[i]t seems that we have long reached an era where the primary focus of the application of [artificial intelligence] in the domain of games is not [solely] on agents and NPC behaviors. The focus has, instead, started to shift towards interweaving game design and game technology by viewing the role of [artificial intelligence] holistically and integrating aspects of procedural content generation and player modeling within the very notion of game [artificial intelligence]" [YT18, p. 15].
So instead of simply focusing on the behavioral side, research is shifting more towards enhancing the overall experience while also making the production of games easier and faster. This includes the creation of stories, levels, music as well as balancing and testing the difficulty of the games or things as simple as smoothing certain controls like the camera movement and placement [BCCT20, 332-333].

3. Artificial Intelligence Methods and Variations commonly used in Modern Day Video Games

3.1. Opponent and Character Artificial Intelligence

As already described in chapter 2.3, many possibilities have opened up over the research period for modeling artificial intelligence and using it in games. The best-known strategies arguably represent so-called "ad-hoc behavior authoring methods". These include *Finite State Machines* as well as *Behavior Trees* [YT18, p. 32].
Due to their simplicity and flexibility in creation as well as in testing and debugging, they are very popular in artificial intelligence development, especially in the role-playing genre, where the creation of many different enemies and behaviors is common.

The main difference between the two is that a Behavior Tree refers to the modeling of decision-making processes of a virtual character in the game world, whereas Finite State Machines describe the behavior of a character by means of different states between which it can switch [MH17, p. 172]. A more detailed distinction and explanation of the techniques underlying these methods is given in the following chapters.

3.1.1. Finite State Machines

The task of a *Finite State Machine* is to model the current state a virtual character is in. However, it is also possible to model, for example, the current game state or the state of the world within a Finite State Machine. The word "finite" is very important in this context, as it symbolizes that only a finite set of possible states which a system can be may exist at a time [JW01, p. 1355][Vya20]. Possible examples of states a virtual character can be in are *walking*, *attacking* or *patrolling*.

Visually, this Finite State Machines are directed graphs where the nodes represent the different states. These nodes are then connected by so-called "transitions", allowing the entity to switch from one state to another. However, the direction in which the transition points is important as it is only possible to switch into that direction, not vice versa.
Whether or not a transition is performed, is determined by set conditions for the transition. Those conditions could require certain parameters to take an expected value or certain triggers to be set. Through the use of random numbers or *fuzzy logic* a certain degree of unpredictability can be added to this process and thereby the behavior of the entity [Cas02, p. 41].
Simply put, Finite State Machines consist of the following three components:

- "A number of states which store information about a task—e.g., [the current state being] the explore state.
- A number of transitions between states which indicate a state change and are described by a condition that needs to be fulfilled—e.g., if [...] a fire [is] shot, move to the alerted state.
- A set of actions that need to be followed within each state—e.g., while in the explore state move randomly and seek opponents" [YT18, p. 33].

An example for a often used Finite State Machine within the Unity Engine are their Animator components.

As described in the previous section, state machines can also be used to model the current state of a game or a character within the game. "Thus, [a game] could have an NPC who first meets a player and gives the player a quest (for example, state before meeting the player is stateintro, changing to statequest after giving the player information about a quest). Then, after the player finishes the quest, the NPC becomes a shopkeeper and sells the player things at a discount as a reward (stateshopkeep)" [Sch09, p. 81].

3.1.2. Behavior Trees

Behavior Trees have "become the leading [artificial intelligence] technique in the video game industry over the last decade" [BCCT20, p. 330]. They are used to create the behavior of a virtual character by modelling its decision making process.

As its name suggests, Behavior Trees resemble a graph with a tree structure. The tree is shown as in 3.1 with the root node at the top. All further levels of the tree are arranged below this root node. The tree is then read and executed from the top left to the bottom right starting with the root node, though the child nodes of a parent node are always explored first before switching to the next node of the same layer [YT18, p. 34-35][MH17, p. 172].

The nodes represent so-called "tasks", which are all based on the same interface and

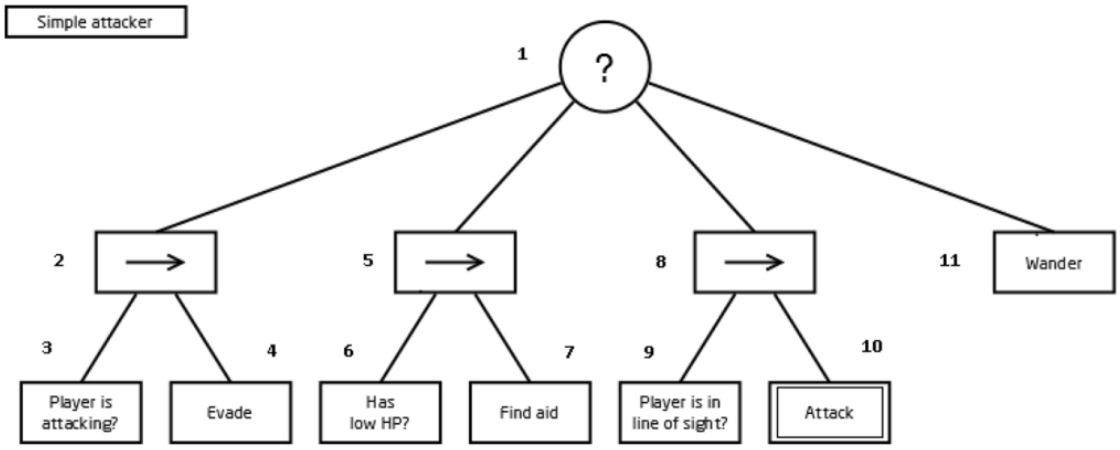


Figure 3.1.: Example for a Behavior Tree labeled as "simple attacker" [MH17, p. 172]

thus have the same API functions. After the execution of such a node or task, it reports a specified status back to its parent node. In most implementations, this status takes one of three values: *Running*, *Success*, and *Failure*. However, some implementations may additionally use an *Error* value [KI"15, p. 511][YT18, p. 34-35].

The following conclusions can be drawn about the execution of the node by means of the return value:

- Success: the task was successfully executed and completed
- Failure: the task was executed completely, but it was not successful with respect to the purpose of execution (for example, because a condition was not met)
- Running: the execution of the task has not yet been completed. The task needs a new execution or waits e.g. for the finished execution of another node or the fulfillment of a certain condition.

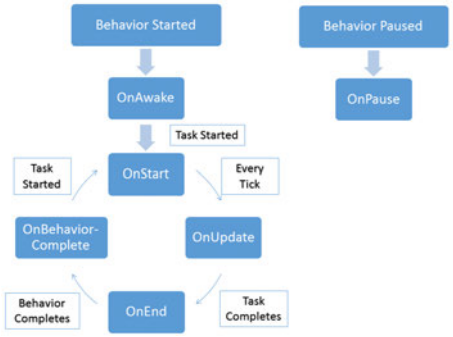


Figure 3.2.: Example for the interface and the execution order of the task API of the behavior designer asset (opsive.com/support/documentation/behavior-designer/tasks/)

- Error: indicates that an error has occurred during the execution of the task.

The root node also returns a value at the end of the execution of the entire Behavior Tree, which allows conclusions to be drawn about the execution of the entire tree [MH17, p. 173].

In addition, tasks are assigned to one of four categories:

- Actions "encapsulate[...] an action to be performed" [MH17, p. 1].
- "Conditions are used to test properties of the environment with boolean-valued functions" [Kl"15, p. 511].
- Composites "act[...] as a control flow component that directs traversal over the tree" [MH17, p. 1].
- "Decorator[s] add [..] complexity to and enhance [...] the capacity of a single child behavior [...] includ[ing] the number of times a child behavior runs or the time given to a child behavior to complete the task" [YT18, p. 34-35]. Additionally, certain types of decorators can be used to manipulate the return status of a child node.

Actions and Conditions tend to act as the ends of a branch of the tree, whereas Decorators and Composites typically have at least one child node. "Note that the statuses within a Behavior Tree are continuously evaluated. Each composite task reacts immediately to changes in its sub-tasks" [Kl"15, p. 512].

However, what this response ultimately looks like depends heavily on the nature of the composite task. Besides *Parallel* tasks, used to execute multiple branches of a tree simultaneously, the two most commonly used composites are *Sequences* and *Selectors*.

Selector nodes succeed, if one of their child nodes returns success. On a failed child node, the next child in order is executed. If there are no more children left, the selector itself returns failure [YT18, p. 34-35]. Therefore, selectors are often "used to provide several alternatives to achieve a common goal. When comparing to logic, selectors can be regarded as an OR-operator" [Kl"15, p. 511].

Sequences on the other hand work like procedures in a regular coding script: they execute all their children from left to right. If the children all return success, the sequence returns success as well. If one of the childnodes fails however, the sequence aborts the execution of the rest of the children and returns failure as well [YT18, p. 34-35]. Sequences "describe a series of tasks in order to achieve higher-level goals. Sequences correspond to the logical AND-operator" [Kl"15, p. 511].

3.2. Meta Artificial Intelligence

Unlike the name might suggest, the term meta artificial intelligence has little to do with the social media company formerly known as *Facebook*. Instead, it describes a form of usage of an artificial intelligence and learning algorithms that use the parts of a game's meta data in order to be able to adjust and therefore balance out difficulty settings at runtime to suit the players skill level and abilities.
Up until now, "in many games, the pace and difficulty of the game is predetermined by the game designer and does not change dynamically during play. The game designer carefully adjusts the parameters such that most players (or a representative player) experiences enjoyment. [...] However, as the skills of players and the game situations are diverse, it is not easy for every player to enjoy exactly the same content" [SM20, p. 3].

In fact, depending on the game, the skill level of two players can be completely different [XWK+17, p. 465]. Nevertheless, the incentive of game designers should be to provide all players with an immersive and fulfilling experience. To achieve this goal, the difficulty of a game should always be kept in the perfect state between over- and underchallenge. This state is also referred to as "flow" [Csi14, p. 136-137]. If the difficulty level is

persistently too high or too low for a player's current skill level, there is a high probability that the player will cease playing the game.
This is where the meta artificial intelligence comes into play: its "basic idea [...] is to dynamically control the contents of the game according to the skill of the player and the game situation, such that various players experience a more interesting game" [SM20, p. 3].

This concept has already been successfully used in video games like *World of Warcraft* or *Left for Dead*. But while World of Warcraft simply adjusted the difficulty of enemies and the game challenge to the players needs, in Left for Dead an artificial intelligence director controlled parts of the entire game world. This includes parts of levels, the spawn locations of enemies and their behavior, as well as sequences of tasks and quests were dynamically generated among others. The use of this technique made for varied and exciting gameplay even after multiple hours of play [BKT14, p. 535-536].
Another example is the companion system in Final Fantasy XV. Here, the developers used the meta artificial intelligence to improve the behavior of the player's allies by making the artificial intelligence "analyze [...] the battle situation and direct [...] an appropriate ally [...] to help the player when in a difficult situation or to follow when the player is running away" [SM20, p. 5].
Even in online games, where players compete against each other, artificial intelligence can help pair players with equally skilled opponents in order to create the fairest competition possible. Similar to matchmaking in chess, elo[1] value methods are used in connection with Bayesian probability models [BKT14, p. 536].

Not only is this kind of artificial intelligence exciting within combat situations. Alternatively, by analyzing the current flow and pacing, it can be used to adjust other content factors such as music, terrain, or the course of the current storyline, thus influencing practically the entire game [SM20, p. 1-2]. By using such technology in conjunction with other methods like procedural content generation, completely customized game experiences can be created.
It is not only the players who benefit from the use of this type of artificial intelligence. Game designers can also benefit from it, as data is collected that shows how the corresponding game needs to be adjusted in order to provide the desired gaming experience to

[1] A system that was originally invented to rank players according to their skill level in playing chess. Nowadays this is a common value used in other competetive gamess, including online games that focus on Player vs. Player combat.

the widest possible range of players. For example, if the game is consistently indicated to be too difficult, game designers can simplify the default state of the game [XWK+17, p. 466].

In order to make these adjustments, hidden meta data is collected, which can be used to calculate a kind of difficulty evaluation score. This score then defines how spawn behavior, enemy behavior and the game's difficulty evolve. The resulting patterns could look like this: "'if the difficulty is 10 then three Type-A enemies appear' and 'if the difficulty is 20 then four Type-A and five Type-B enemies appear.' As long as the player proceeds in the game, increasingly stronger enemies appear, but when the player is defeated, the enemies return to weakness" [SM20, p. 3].
However, a distinction has to be made between different types of players, as "[c]autious players and risk takers [...] behave differently in response to dynamic balancing mechanics" [XWK+17, p. 466].

Evaluation of the score could, for example, be calculated as depending on the damage the player received over a certain period of time as well as on the length of rest periods the player has between two fights.
Additionally, methods exist to draw conclusions about the player's stress perception by determining physiological data such as body temperature or pulse rate and to adapt the game content accordingly [SM20, p. 6]. However, "the majority of [dynamic difficulty adjustment] systems rely upon prediction and intervention as their fundamental strategy" [XWK+17, p. 466].

4. Conducting a Field Test to evaluate the Replay Value of Combat Balancing Methods

4.1. Presentation of the Practical Execution

As mentioned in the introduction chapter, the goal of this bachelor thesis is to find out how to create replay value for action role-playing games through combat between the player and the game's virtual opponents. Most importantly in the course of this study, the replay value has to stem from the artificial intelligence implementation and its balancing, rather than new game design features and adjustments or aesthetically pleasing graphics and special effects. Considering these requirements in combination with the most commonly used tactics to increase the replay value of a game, gradually increasing the difficulty of a game proves to be the best way to achieve this goal.

4.1.1. Combat Difficulty Balancing Approaches used in the Practical Field Test

As stated in chapter 2.2, in order to achieve replayability through difficulty, it is important to keep the player in a constant state of flow, meaning that it is required to remain a constant balance between the skill level of the player and the demand for action, reflexes and strategy. This naturally means that the difficulty has to be increased through the course of a game. The challenge then arises from the stress of constant risk of failure that the player has to overcome by appropriately using their previously acquired skills.

While there are several methods on how to increase the difficulty of a game, the two being most interesting for the purpose of this study are a mathematical and a behavioral approach. The behavioral approach will be the main focus of this study, as it is highly

focusing on alternating the behavior of an enemy which can be done using artificial intelligence implementations. Ultimately, the goal of this thesis is about to examine how replay value of a game can be achieved by using behavioral approaches, hence this is the approach that this thesis examines for its effectiveness in regards to balancing difficulty. The mathematical approach on the other hand, is already commonly used to balance difficulty and has due already proven to be effective in doing so. Therefore, the findings of the test using the mathematical approach will be used as control results to be able to achieve a higher level of precision within the evaluation process for the outcomes of the behavioral approach.

Balancing difficulty by using a mathematical approach

This is the most commonly used technique to balance combat and its difficulty in games. As the name suggests, the balancing with this method is purely mathematical, meaning that the behavior of the enemy remains the same while its stats such as the damage dealt, the attack speed or even its health and defense are increased. This is a relatively easy and rather time efficient way of balancing combat, as it only requires a change of the different stat values and does not require any more design thinking or additional iteration in design or programming. The main challenge for the player lies in gaining the ability to *"read"* the opponent, learn its pattern and react accordingly. Adjusting the different settings results in the following challenges for the player:

- **Increased Damage** ultimately results in the player dying quicker when hit. So the main objective for the player will be to evade the attacks of their opponent. Making hits more unforgiving results in players becoming more cautious with their actions, which at best results in tactical alternation of defensive and aggressive behavior.
- **Higher attack speed** or quicker actions of the virtual character in general will in return demand for higher reflexes of the player in order to be able to evade attacks and land a blow. If smartly paired with increased damage, higher attack speed will result in an especially tough challenge for the player.
- **Enhanced health and defense** will result in the fight becoming longer due to the endurance of the virtual opponent. Usually, the more important and tougher encounters such as bosses or mini-bosses will also have significantly more health-points than *"normal"* encounters that are scattered everywhere in the world.

> The longer the fight lasts, the more endurance is required from a player . Of course, a longer fight doesn't automatically make it more challenging, so increasing an opponent's health and defense should be done in combination with increasing at least one of the first two stats mentioned, if possible.

Clearly, using the mathematical approach will lead to increased stats but will not change the behavior of the opponent. Hence, it is heavily tied with the aspect of learning, so the replay value in this case stems from the fact that the player has to learn the pattern of the enemy and form a plan on how to react in every given situation. The satisfaction of the player stems from the superiority due to tactical knowledge and being able to predict a move. This approach has proven to be highly efficient and is oftentimes used when creating varied difficulty settings for a game.

Diablo 3: Reaper of Souls - Difficulty Overview

(Patch 2.4.1 - Season/Era 6)

Difficulty Setting	Normal	Hard	Expert	Master	T1	T2	T3	T4	T5	T6	T7	T8	T9	TX
Monster Health	100%	200%	320%	512%	819%	1.311%	2.097%	3.355%	5.369%	8.590%	18.985%	41.625%	91.260%	200.082%
Monster Damage	100%	130%	189%	273%	396%	575%	833%	1.208%	1.752%	2.540%	3.604%	5.097%	7.208%	10.194%
+ XP	0%	75%	100%	200%	300%	400%	550%	800%	1.150%	1.600%	1.900%	2.300%	2.750%	3.300%
+ Gold Find	0%	75%	100%	200%	300%	400%	550%	800%	1.150%	1.600%	1.700%	1.800%	1.900%	2.000%
Legendary drop	0%	0%	0%	0%	15%	32%	52%	75%	101%	131%	165% *	207% *	264% *	323% *
Legendary drop (rift)	25%	25%	25%	25%	44%	65%	90%	119%	151%	189%	236% *	295% *	369% *	461% *
Death's Breath	15%	18%	21%	25%	31%	37%	44%	53%	64%	75%	90%	2x 15%	2x25%	2x 50%
Greater Rift Key	50%	50%	50%	50%	55%	60%	70%	80%	90%	100%	2x 5%	2x 15%	2x 25%	2x 50%
Horadric Cache materials	2 (1) **	2 (1) **	2	2	3	3	3	3	3	3	4	4	4	5
Horadric Cache legendaries	10%	10%	10%	10%	10%	50%	60%	75%	90%	100%	2x 5% *	2x 15% *	2x 25%	2x 50%
Keywarden Machine drop	-	-	-	-	25%	28%	33%	38%	43%	50%	55% *	60% *	65% *	70% *
Uber Organ drop	-	-	-	-	100%	2x 10%	2x 20%	2x 30%	2x 40%	2x 50%	2x 100%	3x 10%	3x 25%	3x 50%

Explanation: 2x 50% means there is a 50% chance for a second drop (and 100% for one drop)
* Note: Guesstimates - this data is an estimate based on data gathered by the community.
** Note: Bounty caches on lower difficulties seems to yield less materials. More data needed.
Sources: http://us.battle.net/d3/en/game/guide/gameplay/game-difficulty#general-changes
Data gathered by the Diablo community on Diablofans and Reddit
Feedback to: Bagstone at DiabloFans.com

DIABLO FANS.com

Figure 4.1.: How the different difficulty settings affect the stats and rewards of enemies in Diablo 3 - Reaper of Souls (diablo.fandom.com/wiki/Difficulty)

Balancing difficulty by using behavioral variety

Typically, the alternation of the behavior of an enemy is mostly used in boss encounters or other significant encounters and would not normally appear among smaller monsters and enemies. In comparison to the mathematical approach, it is costlier to implement, as it requires more design thinking, programmatic effort as well as testing for issues.

In case of larger enemy encounters, shifts in the behavior will mostly consist of different

"phases", triggered after a certain event that is usually linked to the progress of the fight. This event could either be the enemy dropping to a certain amount of hit points, killing off other smaller enemies in the area, pulling levers or destroying something in the environment etc. The phases typically manifest in the form of alternate attack patterns or the enhancement of a previous move pool by e.g. increasing the size of the hit boxes.

Switching the behavior of an enemy will serve as a kind of *reset* for the fight in terms of learning patterns and therefore its unpredictability, which is why this approach is mostly linked to the aspect of *variety* and adaption to the different circumstances rather than the typical *learning* aspect that the mathematical approach is addressed. When implemented correctly, alternating the behavior and move pool by assigning probabilities to different moves will even result in the enemy having no distinguishable fighting pattern at all.

Limiting the aspects to be investigated in order to increase the replay value

As already mentioned in the description of the balancing methods, the prototype focuses on increasing the difficulty on the one hand by increasing values and adjusting behavior patterns on the other hand.
The difficulty of the player in the mathematical approach stems mainly from the fact that the player has to try to take as little damage as possible. He achieves this only if he can counteract the opponent's behavior as well as possible, for example by avoiding the opponent's attacks. The challenge in this case is to be able to *read* the opponent and to recognize their behavior patterns. In this case the aspect of learning is the primary objective.
The behavioral approach on the other hand plays a lot with the unpredictability of the opponent's behavior. In this approach, the player has to adapt to the opponent again and again by constantly adjusting his behavior and behavioral patterns. So the challenge in this case is not to read the opponent and learn their behavior, but rather to be able to adapt to different, sometimes randomly driven, behavior patterns. This approach therefore applies to the Variety and Randomness aspect of replayability.

The other three aspects that can contribute to increasing the replay value of a game are not investigated in detail over the course of this field test.

The Social Aspect of the game seems particularly inappropriate for this study because it is a single player mode. The goal to increase the replay value with simple game design mechanics and graphical means is also in contrast to the motivation by creating a memorable Experience or even Nostalgia.
Goals and Completion also seem less interesting due to decision to use the simplest possible game mechanics. While the player's goal is to defeat the enemy of each difficulty level, this motivation is intrinsic, meaning it comes from the player themselves, rather than goals set by the game. The player does not receive rewards in the form of items, nor are score points or other scoring features provided. Although a prototype could possibly be adapted to provide such goals for the player, this would possibly distort the result of the test, making this aspect seem rather insignificant and of little interest for the study.

4.1.2. Procedure of the practical field test

In the field study conducted in this thesis, the aim is to investigate the subjects' or players' reactions and thoughts about the experienced difficulty and their willingness to replay a game based on one of two prototypes consisting of a simple game scene featuring combat between a player character and an artificial opponent. The provided prototype uses either a mathematical or a behavioral approach to gradually increase the difficulty of the fight.
The goal of this study is to find out whether or not difficulty achieved by a pure behavioral approach can provide replay value to a game and if it is competitively viable compared to the often used mathematical approach. In a follow up questionnaire the players will be asked to assess the replay value of the respective balancing technique in regards to a whole game being balanced this way or a game providing three different difficulty settings based on those techniques. Additionally, players will be asked to give their thoughts on meta artificial intelligence and adaptive difficulty settings.

Structure of the Prototypes

Each prototoype features three 3D scenes including a controllable player character and a virtual opponent steered by the artificial intelligence. The first scenes of each prototype are identical and lay the foundation for the survey. In the following scenes however, the

Table 4.1.: How the difficulty in each prototype will be adjusted and balanced

approach	Mathematical	Behavioral
Stage 1	• regular hit and chase behavior • regular damage (30) • regular health (500) • regular attack speed and time between attacks (3-5 seconds)	
Stage 2	• 200% damage • 150% health • 80% time between attacks	• alternating aggressive (attacking and pursuing the player) and defensive (dodging) behavior
Stage 3	• 2 Hit-K.O. for player • 200% health • 50% time between attacks	• faster alternating aggressive and defensive behavior • dodge roll to get close to player more quickly • waiting for player to finish dodge before attacking

difficulty is raised using mathematical adjustments on one and behavioral adjustments on the other prototype as portrayed in table 4.1. Both prototype's artificial intelligence are implemented using behavior trees as described in chapter 3.1.2. This technique has proven to be efficient and well used throughout many games of the role-playing and action role-playing genre, which is why this implementation technique was chosen to be the foundation for this field study.

As combat and the implementation of the artificial intelligence is the main focus of the prototypes, they only contain very basic game design mechanics. Therefore, the move pool for the player and the virtual opponent will be limited to a set of basic actions:

- Horizontal Movement
- Dodge-roll
- 3-Hit Combo with a 30% chance of stunning the target with a successful hit

Additionally, the player will be able to mark the opponent as a target in order to be able to lock the camera.

Graphics wise, the prototypes only use geometric shapes and simple human models and

animations for the characters as well as little to no special effects or sounds. Likewise, the level design of the stage is kept very primitive and consists of a square shaped stage with walls around so that neither the virtual opponent nor the player can fall off the stage. This level structure is the same for all stages across both prototypes.
Everything concerning loot and rewards for the player will be disregarded for the purpose of the study. The incentive for the player to re-battle the opponent should be the fun in the fight and combat, not the desired reward.

Conduction of the field test

The goal of the study is to examine and compare the replay value of the two prototypes. Since the replay value of both of them is based on their difficulty and - especially the mathematically balanced prototype - the learning aspect, it would diminish the significance of the results when letting the subjects play each prototype. Therefore, test subjects are divided into two equal groups with each group playing only one of the two prototypes.

In order to find suitable test subjects, volunteers are asked to fill out a questionnaire (see appendix A). This questionnaire consists of questions about their usual gaming as well as replay habits. Among others, this includes questions such as the preferred game genres and titles as well as overall importance of different factors such as graphics and content.
These questions are intended to aid in categorizing the test subjects so that the two prototypes can be distributed as evenly as possible among the different types of players. In order to be able to draw conclusions in regards to possible target groups and general characteristics of the players, test subjects are asked to provide a password or token in order to be able to match answers from the initial and the follow-up questionnaire as well as recorded testing data.

After the test persons have been divided into groups, they are each sent one of the two prototypes. They are then asked to play the prototype for as long as they want and feel entertained. While playing, data about the respective player behavior is generated and stored. This includes the duration of the overall playtime as well as the time spent in each stage. Additionally, the number of deaths and replays of the individual scenes will be analysed. After playing, the subjects are asked to fill out a follow-up questionnaire

(see appendix B). Subsequently, the filled in questionnaire together with the collected data is sent back to the author for evaluation.

4.2. Implementation of the Practical Test

The technical implementation of the two prototypes is done using the game development engine *Unity*. The programming language used is C#. In addition, changes and progress during implementation and development were stored in a repository using the versioning tool GitLab.

Assets used from the Unity Asset Store

The prototypes are supposed to consist of simple game mechanics and simple graphical elements. Despite this apparent simplicity, there are a lot of components that play a part when creating a combat situation that feels good to the player. Especially the implementation of a smooth-feeling character controller is already anything but trivial. In addition to this came the requirements of an artificial intelligence as well as a combat system through which the player and opponent can interact with each other.

Since implementing these features from scratch would exceed the scope of the bachelor thesis' time frame, ready-made assets from *Unity's* asset store are used where possible. In the case of the character controllers and a combat system, the chosen asset is the *"Third Person Controller - Melee Combat Template"* from Invector[1]. This asset provides most of the functionality needed for the prototypes: for the player character this means horizontal movement, an attack combo as well as a dodge roll and the lock-on function for targeting enemies.
The asset also provides additional movement functionalities such as jumping or sprinting. Also, with regard to the combat system provided, the asset offers further functionality such as heavy attacks, blocking as well as a stamina and inventory system. However, as described in chapter 4.1.2, the combat is supposed to be rather simple in order to better determine whether or not a game can have replay value with an elemental combat

[1]Invector Third Person Controller - Melee Combat Template, https://assetstore.unity.com/packages/tools/game-toolkits/invector-third-person-controller-melee-combat-template-44227

system. That is why these additional functions were disabled for the prototype.

In addition, Invector's asset also provides functionalities for simple artificial intelligence. These include both a companion system and simple melee opponents that can be used as simple enemies or even bosses and can be tested in a demo level. All variables relevant to players or virtual characters, such as health points, damage, speed, etc., can be easily adjusted via the *Unity* inspector. However, in order to have more control over the behavior of the artificial intelligence, the *"Behavior Designer"* asset from Opsive[2] was used instead of solely relying on the provided scripts.
This asset provides the possibility to easily create artificial intelligence behavior in the same fashion as described in chapter 3.1.2. The behavior trees can be modelled with the use of an easy to handle graphical user interface and thereafter stored as external instances which can then simultaneously be used on various prefabs. Additionally, the tool can be fully customized as Tasks can be added by simply inheriting from the Action or Conditional classes.

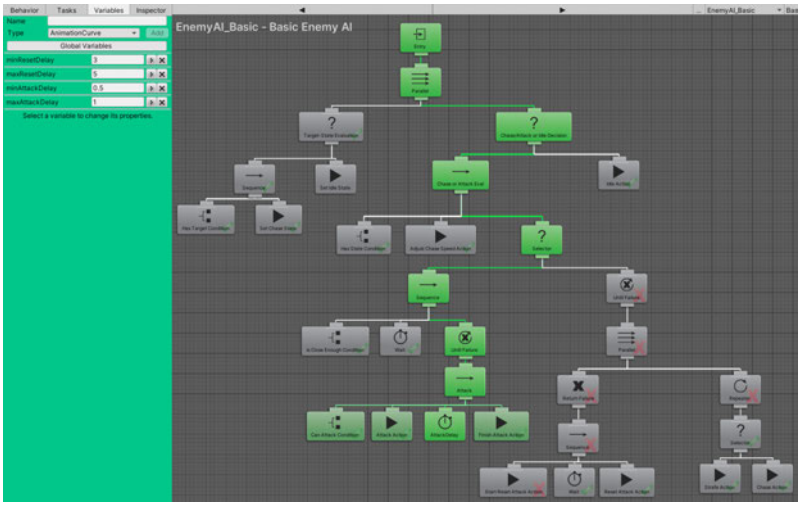


Figure 4.2.: Behavior tree with visual feedback towards execution order and node results at runtime

[2]Behavior Designer - Behavior Trees for Everyone, https://assetstore.unity.com/packages/tools/visual-scripting/behavior-designer-behavior-trees-for-everyone-15277

Structure and implementation of artificial intelligence behavior for the prototype

The tree-like structure, as well as the flexibility and simplicity of adapting the tree and the visual feedback of the nodes[3], make the Behavior Designer a comparatively simple way to realize the behavior of virtual characters. Furthermore, the graphical interface provides an easy way to trace steps of the AI at run-time and debug possible erroneous behavior.
In the case of the prototypes developed within the bachelor thesis, the script for controlling simple melee enemies from the Invector asset was split into individual tasks that inherited either from the action or conditional classes. Afterwards, the behavior of Invector's simple melee opponent was remodelled within the tree structure of the Behavior Designer. After finishing this process, the thereby created behavior tree was used for the behavior of the enemies in the mathematically balanced approach as well as the first stage of the behaviorally balanced approach and lay the foundation for the rest of the required opponents.

As for the mathematically balanced prototype, all that had to be done after creating

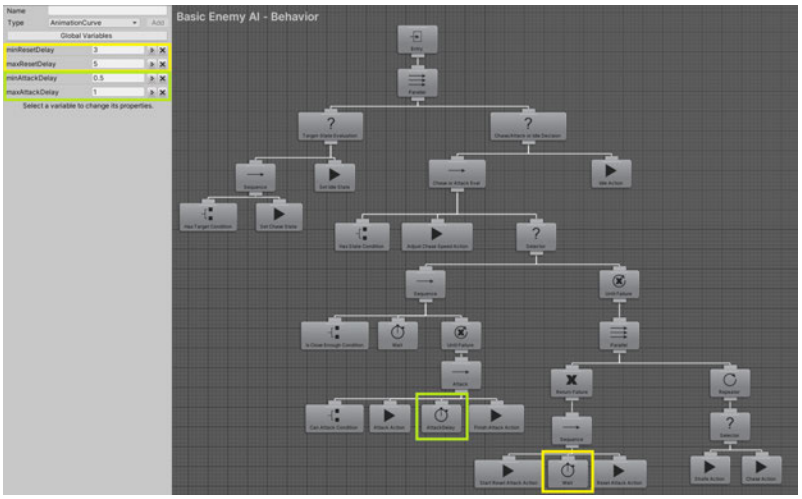


Figure 4.3.: Behavior tree used for the mathematical balancing prototype and as the basis for the behavioral balancing prototype

[3]for example, a check mark at the node for Success or a red X for Failure, as well as the green coloring of the currently executed nodes and paths within the tree, as can be seen in the graphic 4.2 at run-time

the basic behavior was to adjust the corresponding values of the opponents, such as their health points, their damage dealt, and the time between attacks.

The behavior tree provides the possibility to add variables that can then be used to create to alternate the behavior of different enemies even when they use the same behavior tree. In the case of this prototype and the basic behavior, this comes down to the time between individual attacks (here minimum and maximum value of the attack delay) or between the reset of attacks and thus the restart of an attack combo (here minimum and maximum value of the reset delay).
Both values are used within the color-coded *Wait* action seen in 4.3. The actual wait time within the node is then randomized between the respective maximum and minimum value. This randomization is used to at least partially provide a sense of *unpredictability* even for the mathematical approach.

For the behaviorally balanced prototype, the "basic behavior tree" had to be further

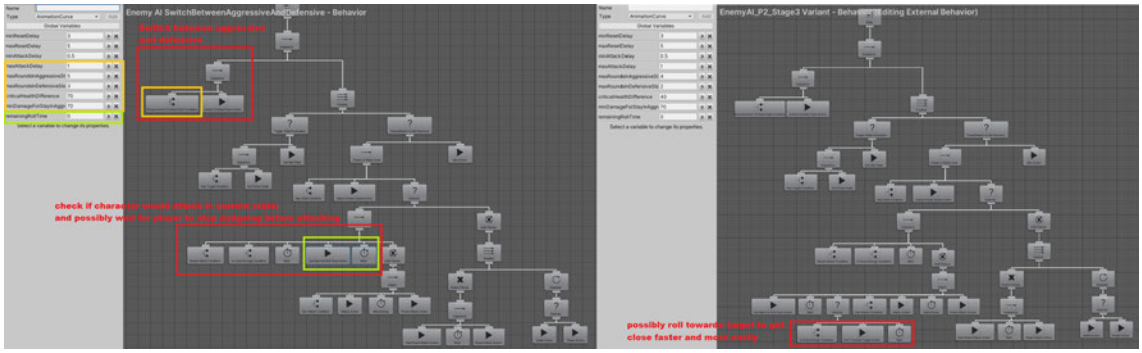


Figure 4.4.: Expansion of the basic behavior tree in 4.3 for second phase enemy (left) and third phase enemy (right)

extended by several functions. The main difference between the second and first phase is the alternation between defensive and aggressive behavior. As the state is relevant for the behavior of the virtual character, this state has to be re-evaluated at the beginning of each run of the Behavior Tree (see the marked red area in the top-left corner of the left behavior tree in 4.4).
Whether or not the artificial intelligence is supposed to change between states is on the one hand determined by how many behavior tree evaluations the virtual character has already been within a certain state and on the other hand, how much damage was dealt either to it or by it. The status mainly influences the behavior of the virtual character in

terms of using its attack or evading incoming attacks. Thus, an attack in the aggressive state is much more likely than in the defensive state and likewise, the virtual character is more likely to try to evade an attack by the player when in defensive mode.
The former is evaluated by using a conditional node before starting the attack cycle, while the latter is determined within the *TakeDamage* function by the virtual character as can be seen in 4.5.

For the third phase, the opponent is also given the ability to quickly close in on the

```
public override float ChanceToRoll {
    get {

        switch (currentCombatStyle) {

            case CombatStyle.Aggressive:
                return combatBehaviorAggressive.chanceForRollInThisState;
            case CombatStyle.Defensive:
                return combatBehaviorDefensive.chanceForRollInThisState;
            default:
                return chanceToRoll;
        }
    }
```

```
public override void TakeDamage(vDamage damage)
{
    // ignore damage if the character is rolling, dead or the animator is disable
    if (isRolling || currentHealth <= 0 || !animator.enabled)
    {
        return;
    }

    if (!damage.ignoreDefense && !actions && CheckChanceToRoll())
    {
        return;
    }

    base.TakeDamage(damage);
}
```

Figure 4.5.: Evaluation of the chance to dodge incoming damage by the player. The probability for the evasion is dependent on the current state of the virtual character and increases when in defensive state, while decreasing when in aggressive state

player by dashing towards them, if the player initially was too far away to be attacked. Additionally, both stage two and three opponents have the possibility to time their attacks so that they are not starting an attack cycle if the player is currently rolling and thereby evading attacks. Instead, the artificial intelligence waits for the player to be done with the dodge roll, before starting the attack cycle. The areas of the behavior tree marked in red in the 4.2 graphic show the respective differences or extensions to the predecessor.

After the different enemy levels were ready and the behavior implementation was done, a simple arena was created using ProBuilder. Afterwards, the respective scenes were linked together. The prototype was then extended to include a “login screen“ using the access token specified by the players. Furthermore, a distinction between the two prototype approaches was added, as well as the functionality to save game data. The saved game data includes the prototype played, the time within a scene, the order in which scenes were played, and whether the player or the opponent won as can be seen in the tables in appendix C.

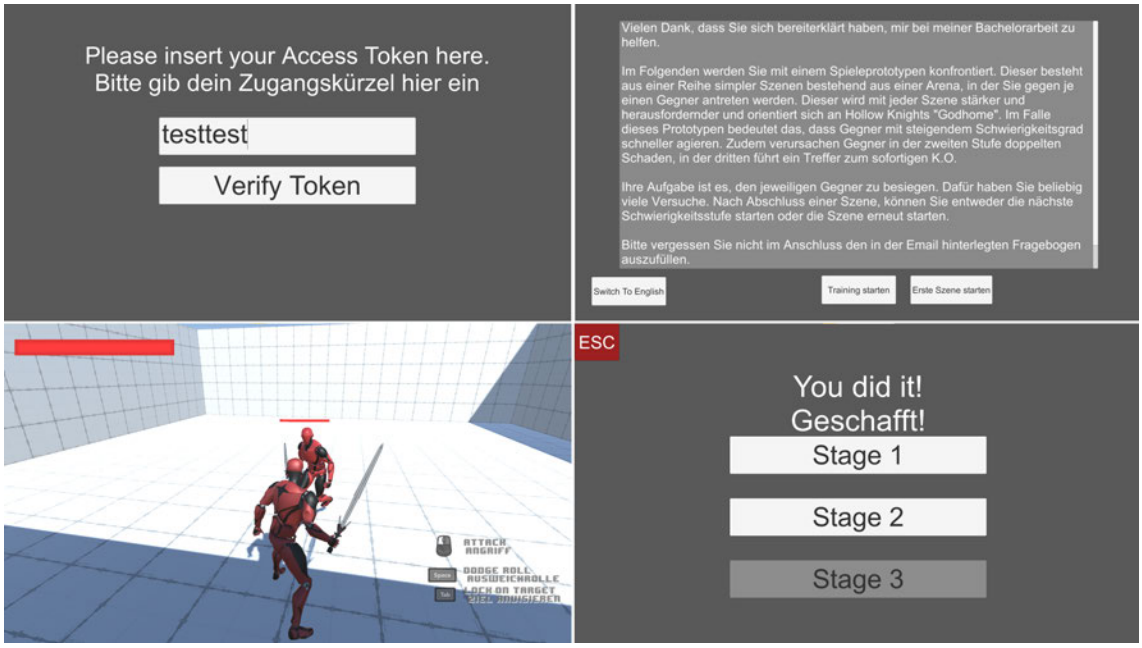


Figure 4.6.: Game scenes of the finished prototype (from top-left to bottom-right: login with access token, disclaimer text, game scene, scene selection after combat

In the end, both prototypes were combined within one build. The distinction with which prototype the player is confronted is made on the basis of the access token, which they must enter in the login scene. This is followed by a short explanation of what awaits the player in the following. Afterwards they can either start a test scene with dummies to get to know the controls of the game or dive straight into combat and the respective stages. After the death of either the opponent or the player, they can choose which of the scenes to play next, though the next difficulty will only unlock once the player managed to defeat the opponent.

4.3. Evaluation of the Practical Test

As part of the practical test, volunteers were initially asked to register as test subjects by means of a questionnaire. A total of 40 volunteers registered this way, which were subsequently divided into two equal groups to test either the mathematical or the behavioral prototype. After the playtest, they were asked to give feedback on it and

on the topic of meta artificial intelligence. This again happened in the form of a follow-up questionnaire. Additionally, the test subjects were asked to submit a json file that consisted of the data that was collected during playtime.
Regarding the test and follow-up questionnaire, a total of 33 responses and 31 game data were collected and submitted. Of these, 17 answers referred to the mathematical prototype and 16 answers to the behavioral prototype.

The registration questionnaire

A questionnaire was used to enroll the volunteers as test subjects for the practical test. In addition to contact information, the questionnaire also contained general questions about gaming preferences and their classic gaming behavior.
The result showed that the role-playing genre is the most popular genre among the test

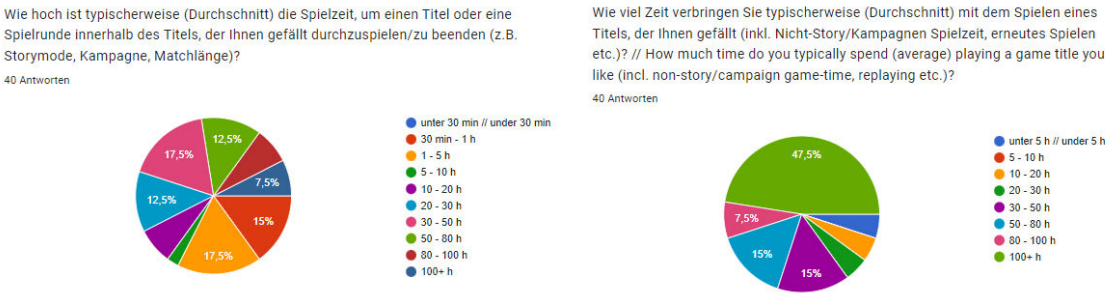


Figure 4.7.: Comparison of the usual time needed to complete a game in comparison with the time actually spent playing these games

subjects with a total of 77.5% of the votes. This was followed by the adventure genre (42.5%), real-time strategy (35%), shooter (30%) and survival (27.5%).
The average playing time to complete a normal story or campaign mode or a game round was between 28.6 and 40.3 hours. The most frequently mentioned game times were between 30 and 50 game hours or 1 and 5 hours. Still, there was an obvious willingness to spend more game time beyond the completion of such a mode or round. Nearly 50% stated, they spend an average of more than 100 hours in total playing their favorite games. On the whole, this results in a playing time of about 67.25 to 77.25 hours or more, which is about twice the playing time of the initially stated average time it takes to finish the respective title.

The willingness to play and replay a game for a longer period of time is also reflected in the next question: only 5% of respondents indicated that they had never replayed a game. More than 75%, on the other hand, stated that they had already played through a game again, a further 7.5% had repeated at least parts or chapters of a game, while the remaining 10% had at least once begun to play a game again.
In this context, it was not at all important what the genre composition or expected playing time of the respective games were. From games with comparatively short rounds like *Age of Empires* or even *Osu* to games with long story campaigns like *The Witcher 3*, *Dragon Age*, *Final Fantasy* or *Gothic* pretty much everything was mentioned.

As seen in 4.8, the main motivations for replaying are to re-experience a story or

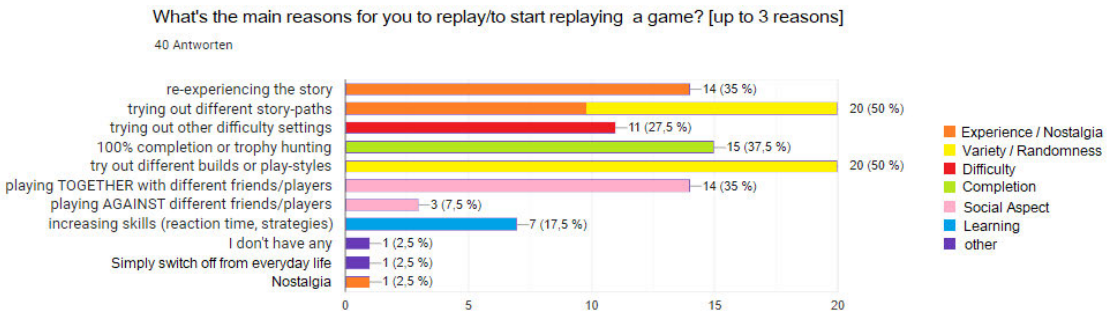


Figure 4.8.: Reasons for replaying a game including classification into the six categories mentioned in chapter 2.2

at least certain parts of it. This also includes experiencing other resulting plotlines that are achieved through different actions and decisions. Players also selected trying out different builds or play styles as a particular incentive to replay a game.
Considering the most frequently chosen motivations, the following ranking of the reasons for replaying mentioned in chapter 2.2 emerges:

1. Variety and Randomness
2. Experience and Nostalgia
3. Goals and Completion
4. Social Aspect
5. Difficulty

6. Learning

The most important factors for choosing which game to play were as much content as possible to explore as well as fluid and intuitive controls that can be adjusted to the players needs. In terms of playing time, the test subjects predominantly preferred longer, compelling campaign modes and stories to shorter (round-based) games.

Evaluation of the prototypes and the follow-up questionnaire

As already mentioned, the 40 subjects who agreed to be voluntary testers of the prototypes were separated into two even groups of 20 testers each. One group was then given access to the mathematical prototype, the other to the behaviorally balanced prototype. Although the opinions of the subjects overlapped and were similar on many questions regarding both prototypes, significant differences between the results could be observed.

One of the first of these differences was the perceived difficulty of each challenge level, as well as the overall impression of the prototype. As can be seen in graph 4.9, the results for the first difficulty level were almost identical. But already from level two on, clear differences regarding the difficulty could be observed: where the behavioral approach was still rated as easy, the general opinion towards the mathematical approach already went into the assessment "difficult" or even "too difficult".
This discrepancy intensified even further towards the third level of difficulty. While almost 50% of the respondents rated the mathematical approach as "much too difficult", the opinions on the behavioral approach were dominated by "somewhat too easy" (56.3%) and "much too easy" (31.3%).

In terms of overall difficulty, the mathematical prototype scored significantly better than the behavioral prototype. The former was rated with slightly more than 1/3 as "just right" and slightly more than 50% as "a little too hard", whereas the behavioral approach was almost consistently rated as "a little too easy" (43.8%) or even "much too easy" (50%).
When conducting the feedback, the major difficulties within the mathematical approach were mostly due to the controls and the perception of the controls as "too clunky". For example, players of the mathematical approach were significantly more likely to want options such as animation canceling, a lower player response time to input, and a larger moveset of ways to engage in combat. In comparison, the feedback of the behavioral

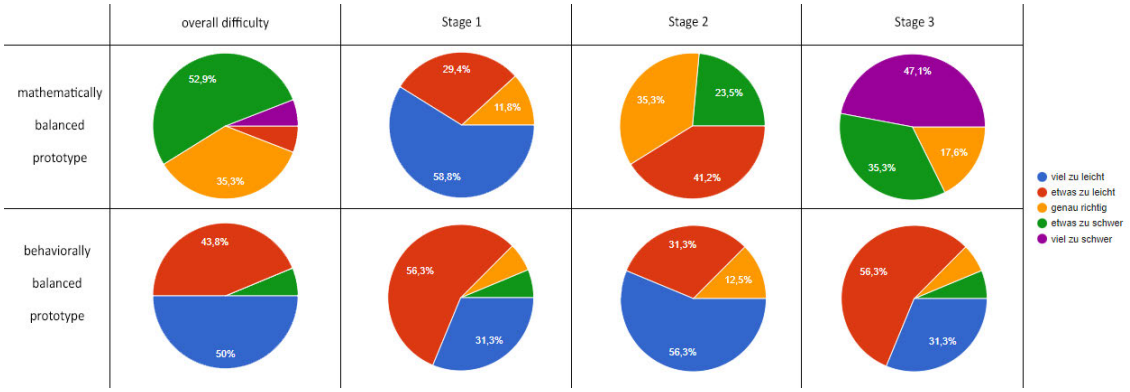


Figure 4.9.: Comparison of the perceived difficulty of the two prototypes in terms of overall challenge as well as difficulty per challenge level

approach was mainly about game design suggestions on how to improve and strengthen the opponent. It was not uncommon for mathematical approaches such as higher damage or health points to be referenced within those suggestions.

This perception of difficulty is also reflected in the recorded game data (see appendix C). In the behavioral approach, almost all players were able to defeat the opponent within the first encounter at all three challenge levels, whereas the number of attempts to defeat the opponent of the same challenge rating in the mathematical approach increased significantly. Already at the second level the number of attempts needed to succeed was 2.78, which increased again to 23.25 attempts needed to succeed at level 3. Comparing the two with one another, the attempt for clearing the third level is about 8 times as many attempts as at level two.

It is especially important to note that only a total of 8 out of 15 individuals whose data could be analyzed were able to successfully pass the third stage, which is only slightly more than 50%. The average number of attempts of those who gave up was about 13.8 attempts.
Due to this significantly higher number of trials that led to success with the mathematical approach, the average time required to play the prototype was almost three times higher than with the behavioral approach. Of course, it would be possible to end the study and conclude that the mathematical approach is better than the behavioral approach in terms of the quality of a game in terms of the cost-value ratio. However, the data and answers obtained show significantly opposite results with regard to the replay value and the willingness to spend more time with a game.

Table 4.2.: Comparison between the number of repetitions of the respective difficulty levels of both prototypes

approach	repeats Stage1	repeats Stage2	repeats Stage3	avg. repeats
mathematical	3	3	2	2,67
behavioral	7	5	8	6,67

As can be seen in the table 4.2, the behavioral approach was repeated by significantly more people after successfully completing each stage compared to the mathematical approach. Also, the actual number of replays beyond the completed attempt is significantly higher on a percentage basis for the behavioral approach.

Considering the results of the follow-up questionnaire with regard to playing behavior and the probability of replaying a game modeled after the respective prototype, similar findings emerge as those obtained from the game data. In these questions, players were asked to imagine the previously played prototype as a finished game. Subsequently, subjects were asked about factors such as their likeliness for purchasing the game and their probable play behavior.

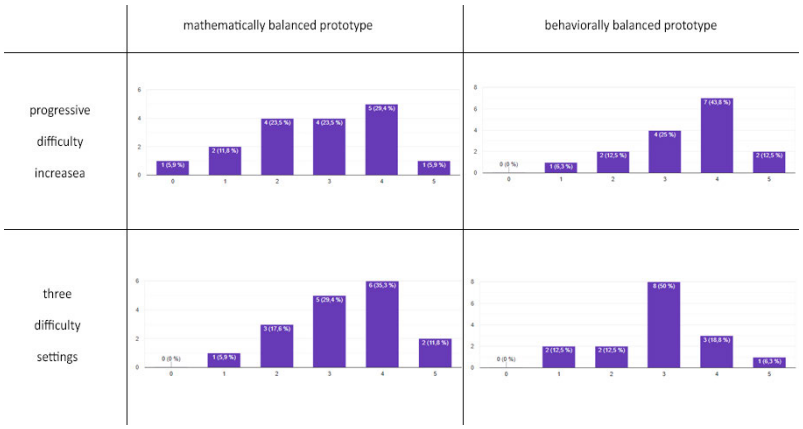


Figure 4.10.: Comparison of the testsubjects' rating for the probability to buy a game that is balanced similar to the previously played prototype

The same questions were asked twice. The difference was that in the first round of questions the respondents were asked to imagine the increase in challenge of the

prototype as a progressive increase in challenge within the course of a game, whereas the second round of questions referred to three different levels of challenge (normal, difficult, pro) between which the players could choose.

As for the question about the probability of purchasing such a game, the behavioral approach with progressive difficulty increase was found to be the most attractive prototype for buying the game. With the mathematical approach, a game with fixed, adjustable difficulties enjoys a significantly higher popularity than the progressive difficulty increase. One reason for this may be that especially the difficulty curve between the second and third level turned out to be too difficult and extreme for most players. In the example with different difficulty modes, players at least have the option to choose what kind of challenge they want to be confronted with.

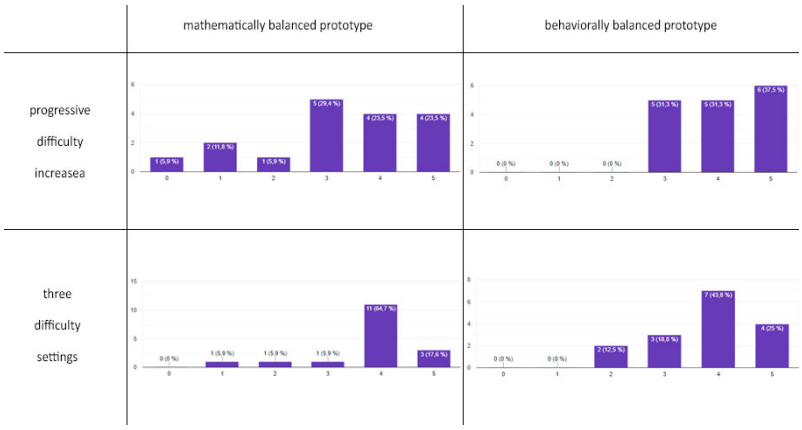


Figure 4.11.: Comparison of the testsubjects' rating for the probability to finish a story or campaign of a game that is balanced similar to the previously played prototype

The behavioral approach with progressive difficulty increase also clearly surpassed the other options when it comes to the question of how likely players are to complete the respective game (see figure 4.11) if it is already in their possession, as well as the probability of continuing to play it or replaying the game after the game has ended (see figure 4.12).

For the majority, the expected playing time that the game would provide was rather

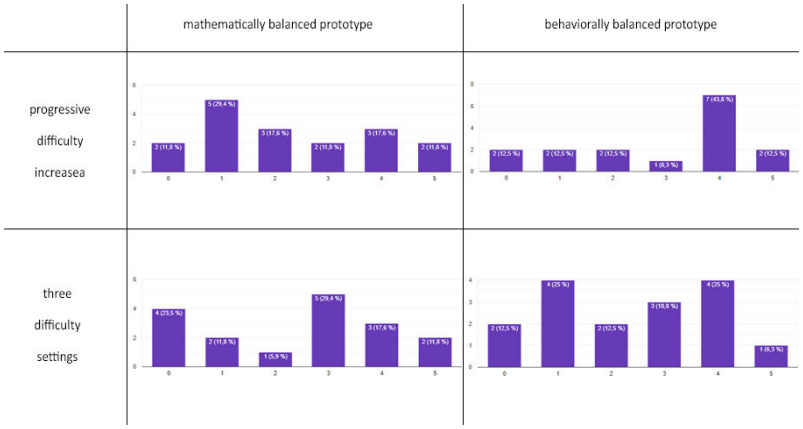


Figure 4.12.: Comparison of the testsubjects' rating for the probability to replay a game that is balanced similar to the previously played prototype

unimportant. However, with regard to the money that the test persons would be willing to spend for the respective game, a significantly higher price classification was made for the progressive difficulty increase with behavioral balancing than with the others.
For a game like that, almost 40% indicated a willingness to spend a price of 40-60€. In the mathematical approach with progressive difficulty increase, on the other hand, the amount of money spent was equally divided between 10 and 19€ and 20 and 29€, and thus only about half of the price indicated in the behavioral approach. Even for the three difficulty levels, at least 1/4 of the respondents would be willing to spend 40-60€ for the game that provides behavioral balancing, whereas for the mathematical share the distribution is relatively similar to the progressive difficulty increase.

As for the genre of the respective game, as seen in B.2.2 and B.3.2, in all four examples the Role-Playing Game genre was very much up front as the preferred game type. In the mathematical prototype with three difficulty levels, the genre received a commitment of almost 65%. In the progressive difficulty increase, the Rogue-Like genre was also strongly represented with about 20% each. In the behavioral approach, the progressive prototype and especially the prototype based on three difficulty levels also showed a clear tendency towards the adventure genre (18.8% and 43.8%, respectively).

For the majority, the most crucial factor for the purchase was a unique game mechanic or a unique combination of different game mechanics. The feedback regarding the changes to the difficulties and the improvement of the prototype to a game that appeals to the test persons, covered several suggestions regarding the game design in terms of the

mechanics provided for the player and the optimization of the controls. Above all, more possibilities for development within the game were desired. Regarding movement and combat mechanics, the following mechanics emerged:

- more control over the player character in terms of canceling attacks or dodge rolls (Animation Cancel)
- Variety of movement and attack options: light and heavy attacks, special abilities, holding the attack button for a heavy charged, jumping, dodging backwards
- Ability to block or parry attacks, instead of just the dodge roll
- visual and auditory feedback within the fight

Considering such feedback, it is apparent that while the design of the combat and the variance of the enemies is significantly important to the longevity of a game, ultimately the game design and the interplay of the world, mechanics, and the virtual characters are significantly more crucial and important to the continued enjoyment and retention of the players in the game and the experience gained from it.

Meta Artificial Intelligence

In addition to feedback on the respective prototype and the data collected through it, the subjects were given a brief introduction and explanation of the concept of "Meta Artificial Intelligence" at the end of the questionnaire. Subsequently, they were asked opinions and judgments on the topic.

The first questions initially related to the general attitude to the topic. Already 54.5% of the respondents stated that they had heard the term "Meta Artificial Intelligence" or even played a game that supports and uses this technology. Initial attitudes toward the topic of adaptive difficulty adjustments were relatively evenly split, but leaned more heavily toward more players wanting this type of balancing in games as can be seen in B.4.3. Similar results were obtained when subjects were asked about their views on the general use of artificial intelligence to create individual games or gaming experiences. This was not only about adaptive difficulty adjustment, but also procedural content generation and technologies that can be used to individualize content.

Regarding the purchase motivation, a majority of the respondents stated that they

would very probably want to buy a game with adaptive difficulty. Only 5 votes of the respondents settled in the ranges of "rather unlikely" to "very unlikely" . Most of the test subjects (42.2%) wanted a price of up to 29€. A little more than 1/5 would even be willing to spend between 40 and 60€ for such a game and thus a more individual gaming experience.

With regard to gameplay behavior, most indicated that they would at least play through and finish a game with adaptive difficulty balancing, but when asked whether players would play through such a game again, the distribution of probability was much more distributed and settled mainly in a kind of neutral agreement between "likely" and "unlikely" as can be seen in B.4.8.

This division is also reflected in the general opinion and feedback on the topic of meta artificial intelligence: although many would like to have a more individual game experience, it still requires certain limits in terms of the difficulty range in which the game then varies.
Particularly on the topic of the survival genre, there were many concerns that a meta artificial intelligence could create a sense of "unfairness". Major concerns regarding this genre were that survival games would destroy the incentive to prepare, as this could be counteracted by the meta artificial intelligence, which would effectively penalize "good play".
Similarly, in the field of fighting and strategy games, concerns have been expressed that the appeal of the game comes from the same basic requirements of the competing parties. The challenge in this type of game is to use one's own skills *better* than the opponent. If one of the two parties (for example the computer) would get an advantage by a superior instance, then it rather gets the same bad taste as being cheated on, as is the case with "rubber-banding" instances that are often used in racing games.

Other major concerns are expressed about whether opponents could be tricked and manipulated more strongly than is already the case in a lot of video games. On the other hand, people are likewise concerned, that the increase in difficulty would be too strong to make the game playable for the player. Here again, there is a plea for a limitation of the meta artificial intelligence or the possibility to turn adaptive challenge adjustments off completely and to be able to manually choose a level of difficulty instead.

5. Summary and Conclusion

The research field of artificial intelligence has made massive technological progress over the past 30 years [BKT14, p. 534]. This is also evident in terms of its development in the field of video games: previously "newest " and "most realistic " graphics and other visual enhancements played a large part in the marketing strategy of many games. Nowadays, those aims to create better graphics have lead to the possibility to create more and better artificial intelligence for games [Sch09, p. 6].

The term "intelligence" describes the ability to acquire knowledge and the ability to utilize it in a variety of contexts [Sch09, p. 2]. The term "artificial intelligence", on the other hand, describes the modeling of such intelligence. Human intelligence usually serves as a model for this.
How this intelligence is achieved is usually irrelevant. More important is the result. Thus, models such as "Behavior Trees" or "State Machines" are technically not considered artificial intelligence according to the actual definition, but are nevertheless referred to as such in mainstream gaming [YT18, p. 7].

Artificial intelligence is used in many areas of the gaming industry today. Probably the most obvious example is virtual characters such as opponents, companions or other non-playable characters. Here, artificial intelligence mainly refers to the behavior modeling of such a character, which can also include pathfinding algorithms or intelligent dialog systems, among others. Artificial intelligence can be used to make battles more exciting, for example by having opponents follow certain strategies and tactics, or by having a horde of opponents organize themselves as a group and act tactically smart.
Other technologies such as meta artificial intelligence further enable the difficulty within a game to be manipulated at runtime and customized to suit the player's ability level in order to preserve the feeling of "Flow" [Csi14, p. 136-137] (see chapter 3.2 for further detail). However, the fields of content generation, auto play-testing and other areas of development are also becoming more and more common research fields for artificial intelligence.

Due to the multitude of possible applications, including content generation, the individualization of the gaming experience and the adjustment of difficulty levels, artificial intelligence can also contribute to significantly increasing the duration of a game or even its replay value and thus the longevity of a game.

The reasons for replaying a game are numerous, but can mostly be assigned to one of the following six categories:

- Experience or Nostaliga
- Goals and Completion
- Social Aspect
- Variety and Randomnesss
- Difficulty
- Learning

[FGLS11, p. 20-21][Kos13, p. 38-42] In games with a combat focus and the corresponding design of the opponent behavior, the replay value of a game can be increased by the correct use of artificial intelligence. This is especially the case in the previously mentioned categories of Difficulty, Learning as well as Variety and Randomness.

Increasing the replay value in these areas is especially relevant to the genre of action role-playing games. The genre is based on the concept of an adventure or a hero's journey with a primary focus on real-time combat, in which skill checks regarding the game mechanics in combination with the player's quick reflexes and hand-eye coordination are required. In most cases, there are many different types of enemies within this genre with various behavioral patterns and abilities. These can be small, "light opponents" as well as large boss monsters [Wri20][Sch09, p. 73].

In this bachelor thesis, two typical balancing approaches for the opponent difficulty of this genre were modeled and subsequently analyzed: a mathematical and a behavioral approach. On the game design side, the resulting prototypes were based on simple mechanics. These consisted of

- horizontal Movement

- camera rotation and lock-on function to target opponents
- three hit combo
- dodge roll
- 30% chance to stun the opponent for a short time after a successful attack

In addition, an opponent was designed using behavior trees, which was balanced using one of two approaches. Specifically, a mathematical approach with a simple increase of values such as health points or damage dealt and a behavioral approach where the behavior of the opponent is adjusted with increasing difficulty and made more unpredictable by adding variance[1] as well as making the opponent's decisions overall smarter.

In the mathematical approach, the main challenge is to *analyze* and *learn* the opponent's behavior in order to respond to them in a faster and more efficient way. In the behavioral approach, on the other hand, the difficulty comes from the ongoing *variance* and, if necessary, *randomization* of the opponent's behavioral patterns. Thus, the challenge in this approach stems from the fact that the player has to permanently adapt to the opponent's changing behavior. If opponents now also adapt their tactical behavior in order to be able to react better to the player, this creates a dynamic that cognitively challenges the player throughout the entire battle, instead of providing the possibility to rely on mere memorization.

Based on the analysis of the prototypes in connection with their attractiveness in terms of purchase motivation, motivation to play-through, and their replay value, the behavioral approach was predominantly shown to be superior to the mathematical approach. However, since mere behavioral adjustment was considered too easy in terms of difficulty, it is advisable to rely on a mixture of both methods. In addition, the written feedback and comments of the test subjects showed that a certain degree of predictability is welcome from time to time and not a bad thing as it also prevents players from burning out quickly [Sch09, p. 37-38].

In conclusion, it can thus be said that the use of artificial intelligence within the gaming sector will increase in the future. This may also involve an interplay of many different implementation strategies. Behavior trees and state machines, have proven to not necessarily be worse than other implementation strategies when done correctly. However,

[1] for more details see the chapter 4.1

other technologies, such as decision making, meta artificial intelligence or instances, can be used to further improve the performance of the virtual characters or the game world in general. When focusing on combat artificial intelligence, the main focus should be on a variance in enemy behavior, though developers can also rely on the classic mathematical way when balancing the opponents' strength.

Bibliography

[Ada01a] Ernest Adams: *Replayability, Part 2: Game Mechanics*, July 2001, URL: https://www.gamedeveloper.com/design/replayability-part-2-game-mechanics, last accessed 15.05.2022.

[Ada01b] Ernest Adams: *Replayability, Part One: Narrative*, May 2001, URL: https://www.gamedeveloper.com/design/replayability-part-one-narrative, last accessed 15.05.2022.

[Bai21] Jon Bailes: *Dark Souls' influence on game design refuses to die - A decade on, From Software's classic continues to inspire a generation of developers*, Nov 2021, URL:https://www.pcgamesn.com/dark-souls-remastered/influence-surge-thymesia-eldest-souls, last accessed 03.01.2022.

[BCCT20] Bruno Bouzy, Tristan Cazenave, Vincent Corruble, Olivier Teytaud: *Artificial Intelligence for Games*, in *A Guided Tour of Artificial Intelligence Research 2: AI Algorithms* (edited by Pierre Marquis, Odile Papini, Henri Prade), p. 313–337, Springer, Cham, 2020.

[BKT14] Christian Bauckhage, Kristian Kersting, Christian Thurau: *Künstliche Intelligenz für Computerspiele - Historische Entwicklung und aktuelle Trends*, *Informatik Spektrum*, Vol. 37, p. 531–538, Dec 2014.

[Byc09] Josh Bycer: *The tenets of replayability*, Dec 2009, URL: https://www.gamasutra.com/blogs/JoshBycer/20090612/1842/The_tenets_of_replayability.php, last accessed 20.05.2022.

[Cas02] Stephen Cass: *Mind games [computer game AI]*, *IEEE Spectrum*, Vol. 39(12), p. 40–44, Dec 2002.

[Csi14] Mihaly Csikszentmihalyi: *Flow and the Foundations of Positive Psychology. The Collected Works of Mihaly Csikszentmihalyi*, Springer, Dordrecht, 2014.

[Elm19] Wilfried Elmenreich: *Short Games - Quickly Made, Quickly Played*, in *Agency, Design, Engineering* (edited by Andreas Rauscher), p. 41–53, Springer VS, Mainz, 2019.

[FGLS11] Timothy Fattesi, Douglas Griesbach, Jonathan Leith, Timothy Shaffer: *Replayability of Video Games*, Researchpaper, Worcester Polytechnic Institute, Worcester, May 2011.

[JW01] Daniel Johnson, Janet Wiles: *Computer games with intelligence*, in *10th IEEE International Conference on Fuzzy Systems. (Cat. No.01CH37297)*, Vol. 2, p. 1355–1358, IEEE, Melbourne, 2001.

[Kl"15] Andreas Klöeckner: *Behavior Trees with Stateful Tasks*, in *Advances in Aerospace Guidance, Navigation and Control* (edited by Joël Bordeneuve-Guibé, Antoine Drouin, Clément Roos), p. 509–519, Springer, Cham, 2015.

[Kos13] Raph Koster: *A Theory Of Fun For Game Design*, O'Reilly Media, Inc., Sebastopol, 2 Ed., 2013.

[Kuf18] Yaroslav Kuflinski: *Artificial Intelligence in Video Games: A Perfect Couple*, Oct 2018, URL:https://www.iflexion.com/blog/ai-in-games, last accessed 15.02.2022.

[MH17] Ryan Marcotte, Howard J. Hamilton: *Behavior Trees for Modelling Artificial Intelligence in Games: A Tutorial*, *The Computer Games Journal*, Vol. 6, p. 171–184, 2017.

[Pet18] Juhana Pettersson: *The Art of Experience*, in *Role, Play, Art - Collected Experiences of Role-Playing* (edited by Föreningen Knutpunkt), p. 101–108, Stifelsen framtidens kultur, Stockholm, 2018.

[Sch09] Brian Schwab: *AI Game Engine Programming*, Course Technology, Boston, 2 Ed., 2009.

[SM20] Daiki Satoi, Yuta Mizuno: *Meta Artificial Intelligence and Artificial Intelligence Director*, in *Encyclopedia of Computer Graphics and Games* (edited by Newton Lee), Springer, Cham, 2020.

[Ste15] Robert J. Sternberg: *Multiple Intelligences in the New Age of Thinking*, in *Handbook of Intelligence: Evolutionary Theory, Historical Perspective,*

and Current Concepts (edited by Sam Goldstein, Dana Princiotta, Jack A. Naglieri), p. 229–241, Springer, New York, 2015.

[SZ04] Katie Salen, Eric Zimmerman: *Rules of Play - Game Design Fundamentals*, MIT Press, Cambridge, 2004.

[Thy14] Lasse Weissenberg Thygesen: *Replayability: A Structural Approach to Players and Computer Games*, Researchpaper, Aalborg University, Aalborg, Jun 2014.

[Vya20] Vedant Vyas: *Artificial Intelligence in Games*, Oct 2020, URL:https://arcanelost.com/artificial-intelligence-in-games/, last accessed 15.02.2022.

[Wri20] Will Wright: *Guide to Video Game Genres - 10 Popular Video Game Types*, Nov 2020, URL: https://www.masterclass.com/articles/guide-to-video-game-genres#10-popular-video-game-genres, last accessed 01.04.2022.

[Wri22] Bryan Writz: *The Complete Guide to Video Game Genres - From Scrollers, Shooters, to Sports*, Mar 2022, URL: https://www.gamedesigning.org/gaming/video-game-genres/, last accessed 01.04.2022.

[XWK+17] Su Xue, Meng Wu, John F. Kolen, Navid Aghdaie, Kazi A. Zaman: *Dynamic Difficulty Adjustment for Maximized Engagement in Digital Games*, *Proceedings of the 26th International Conference on World Wide Web Companion*, p. 465–471, 2017.

[YT18] Georgios N. Yannakakis, Julian Togelius: *Artificial Intelligence and Games*, Springer, Cham, 2018.

[ZD18] José P. Zagal, Sebastian Deterding: *Definitions of "Role-Playing Games"*, in *Role-Playing Game Studies: Transmedia Foundations* (edited by José P. Zagal, Sebastian Deterding), p. 19–51, Routledge, New York, 2018.

Appendix

A. Volunteer registration

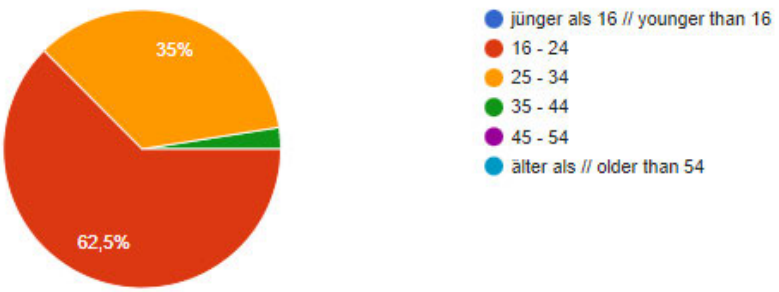
This questionnaire was used to enroll test subjects for the prototype test conducted within the practical study. The information provided here represents the overall results of the survey, including information about the classical gaming behavior of the players as well as questions that give information about their respective player type according to the Barthel-Index. This information can subsequently be used to draw conclusions about any target groups in connection with the responses from the result of the follow-up questionnaire (Appendix B).
In total, 40 responses were submitted.

A.1. Contact Information

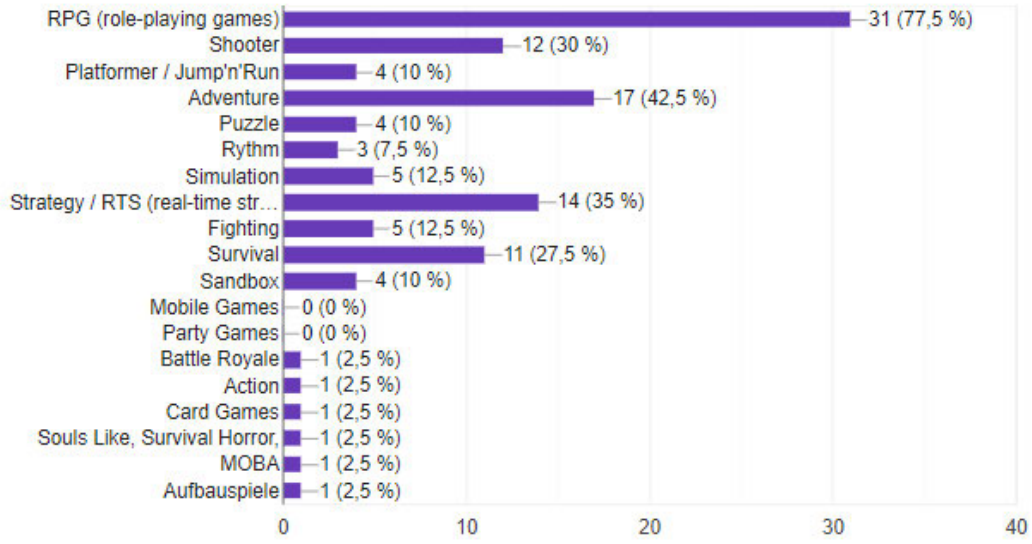
This section was excluded due to the data protection regulations applicable in Germany.

A.2. General Information on the Player

A.2.1. How old are you?



A.2.2. What type of game genre do you usually prefer playing [choose up to 3]?



A.2.3. What are your top three favorite games? (Nr. 1)

- Age of Empires 2
- Ark Survival Evolved
- Assassin's Creed
- Assassin's Creed 2
- Assassin's Creed Valhalla
- Bloodborne
- Dark Souls
- Dead by Daylight
- Dead Cells
- Deadspace
- Destiny 2

- Elden Ring (x2)
- Fallout 3
- Fallout 4
- Faster than Light
- Final Fantasy X
- Final Fantasy XII
- Final Fantasy XV
- Get in the car, loser!
- Gothic
- Hades
- League of Legends (x4)
- Nier Automata (x2)
- Outer Wilds
- Pokemon Smaragd
- Ratchet & Clank
- RimWorld
- Sekiro
- Skyrim
- Smite
- Sonic Heroes
- Subnautica
- The Elder Scrolls III: Morrowind
- The Forest
- The Witcher 3 (x3)

- World of Warcraft

A.2.4. What are your top three favorite games? (Nr. 2)

- Age of Empires 1, 2 und 4
- Assassin's Creed Origins
- Bloodborne
- Call of Duty
- Civilization
- Dark Souls 3
- Destiny 2
- Dungeons and Dragons
- Elite Dangerous
- Far Cry 4
- Final Fantasy 7 Remake Integrade
- Gothic 2
- Highrise City
- Horizon Zero Dawn
- Journey
- League of Legends
- Life is Strange
- Mass Effect
- Metal Gear Solid 4
- Minecraft (x2)
- No Mans Sky

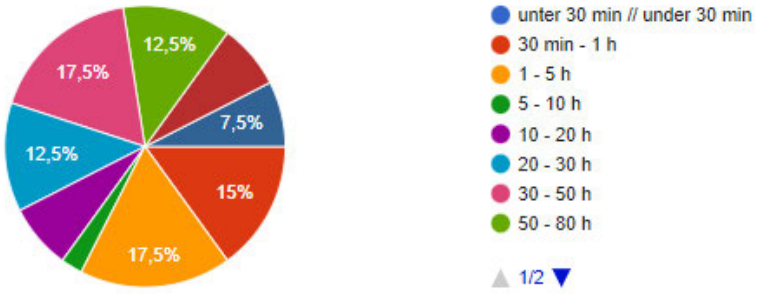
- Odin Sphere: Leifthrasir
- Overwatch
- Paper Mario
- Pokemon
- Pokemon Smaragd
- Quake 1
- Realm of the Mad God
- Red Dead Redemption 2
- Smackdown vs. Raw 2011
- Subnautica
- The last of us 2
- They Are Billions
- Titanfall 2
- Warframe
- Warhammer Online
- War Thunder
- Witcher 3
- Yakuza 0
- Zelda Breath of the Wild

A.2.5. What are your top three favorite games? (Nr. 3)

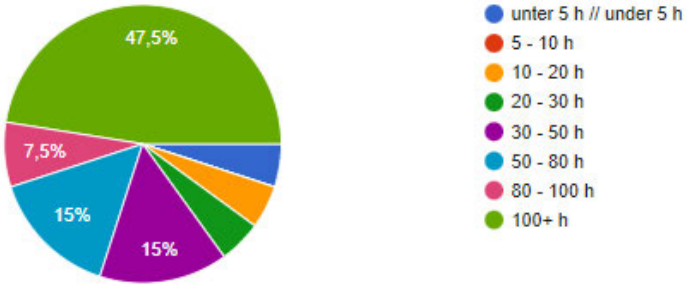
- Anno 1503
- Age of Empires
- Baba is You
- Battlefield 1
- Cities: Skylines
- Clonk 4
- Cuphead
- Divinity: Original Sin 2
- Detroit Become Human
- Dragon Age Origins
- Factorio
- Frostpunk
- Germinal
- God of War (2018)
- Gothic
- Horizon Zero Dawn
- Kenshi
- Kerbal Space Program
- Kingdom Hearts 2
- Legend of Zelda: Ocarina of Time
- Magic: The Gathering / MTG Arena (x2)
- Mass Effect
- Minecraft

- Mini Metro
- Ori and the Blind Forest
- Ori and the Will of the Wisp
- Osu!
- Overwatch
- Paper Mario The thousand year door
- Path of Exile
- Sea of Thieves
- Stray
- Super Mario Sunshine
- Titan Quest
- Tomb Raider
- Uncharted
- Valheim
- Witcher 3 (x2)
- World of Warcraft

A.2.6. What is typically (average) the play time to play through/finish a title or game round within the title you like (e.g. story mode, campaign, match length)?



A.2.7. How much time do you typically spend (average) playing a game title you like (incl. non-story/campaign game-time, replaying etc.)?



A.2.8. Have you ever replayed a game?

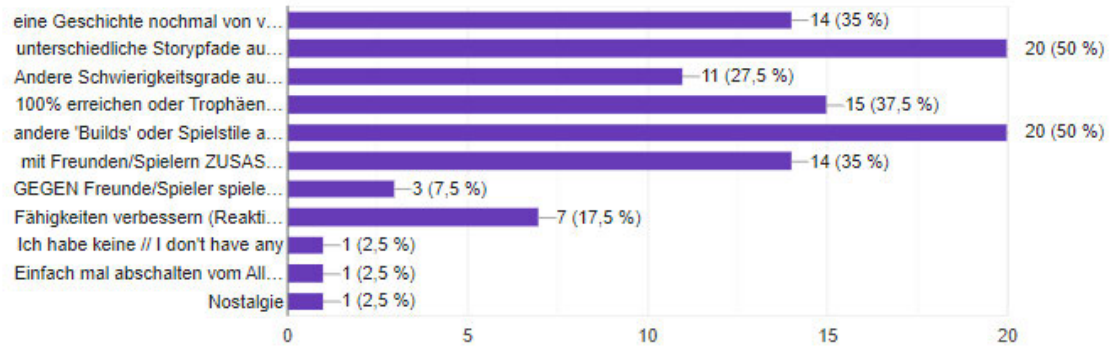
A.2.9. If you've answered the last question with a "yes" or "started": which title(s) was/were it? (name up to 3 titles)

- Age of Empires 3
- Alien Isolation
- Assassin's Creed 2 (x2)
- Battlefield 3
- Beyond Two Souls
- Bioshock 1
- Bloodborne (x2)
- Borderlands 2
- Call of Duty
- Crash day
- Darksouls (x3)
- Dark Souls 3 (x3)
- Dead Space

- Detroit: Become Human
- Diablo 3
- Dragon Age (x2)
- Dragon Age Origins
- Dragon Quest VIII
- Elden Ring
- Factorio
- Fallout 4
- Far Cry 1
- Far Cry 4
- Faster Than Light
- Final Fantasy
- Final Fantasy 7: Remake Integrade (x2)
- Final Fantasy X
- Final Fantasy XII: The Zodiac Age (x2)
- Final Fantasy XV
- Gothic 1, 2 und 3 (x2)
- GRIS
- Hades
- Kingdom Hearts 2
- Life is Strange (x2)
- Mass Effect
- Metal Gear Solid
- Minecraft (x4)

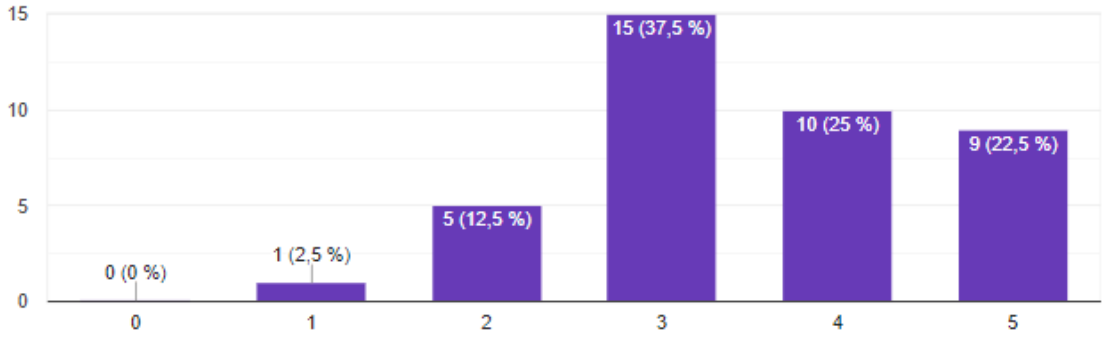
- Nier Automata
- Nier Replicant
- Ori and the Blind Forest
- Ori and the Will of the Wisp
- Paper Mario
- Pokemon (x4)
- Pokemon blau
- Portal
- Quake
- Resident Evil 4
- Sekiro
- Sonic Heroes
- Spore
- Star Wars: Empire at War
- The Witcher 3 (x4)
- The Last of Us
- Titanfall 2
- Torchlight 2
- Titan Quest (x2)
- Undertale (x2)
- Until Dawn
- Valheim
- World of Warcraft (x2)

A.2.10. What's the main reasons for you to replay/to start replaying a game? [up to 3 reasons]

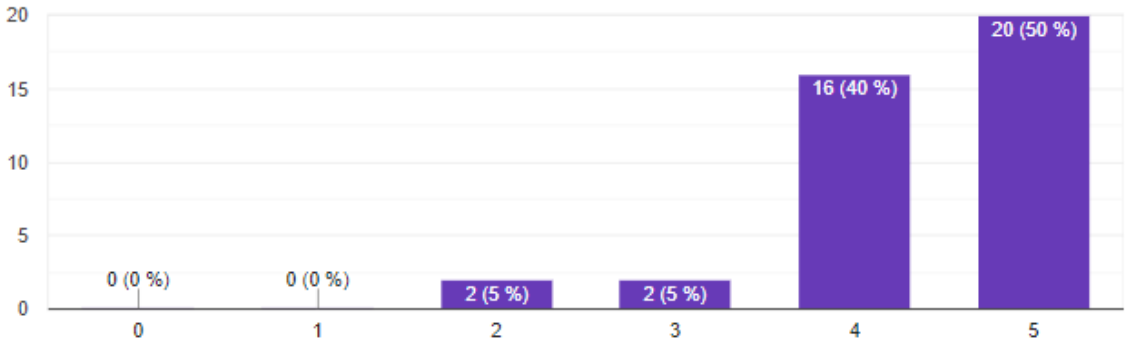


A.3. Player Type Analysis

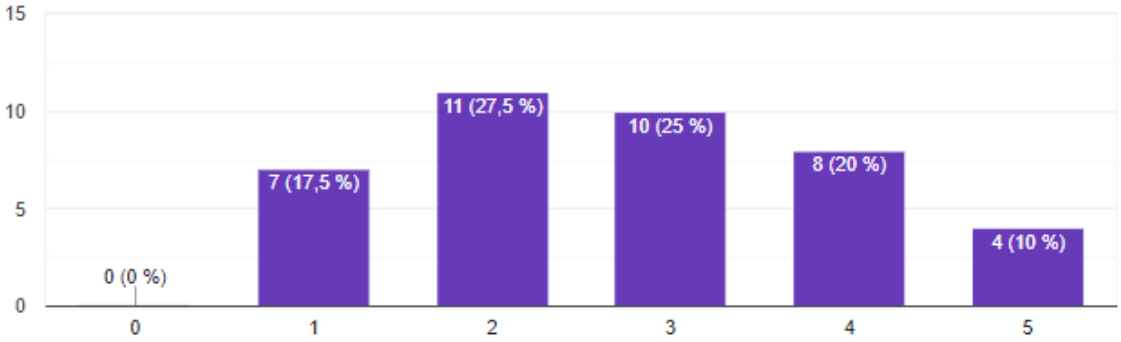
A.3.1. The graphic appearance must be aesthetically pleasing.



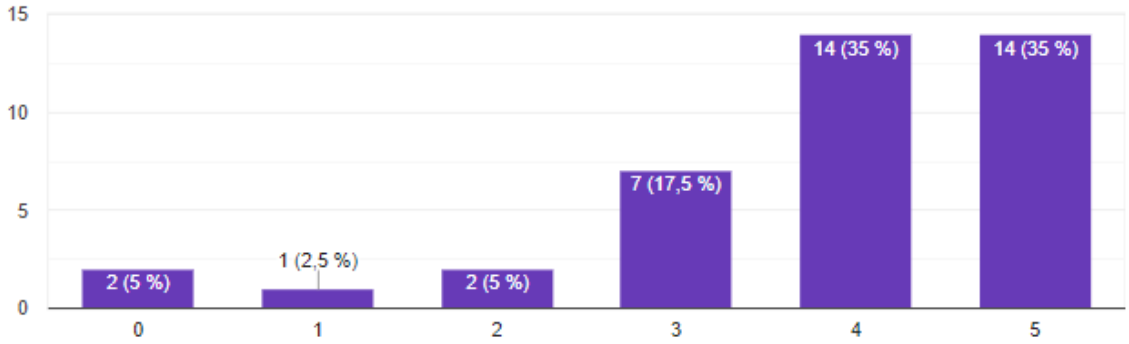
A.3.2. Fluent and intuitive controls are a must-have for me. I need to be able to adjust them to my needs.



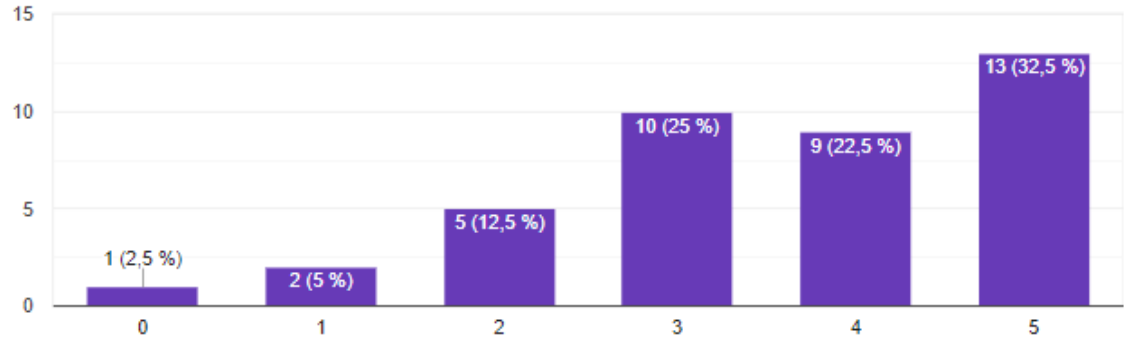
A.3.3. I like fast-paced gameplay. I want to use my quick reflexes.



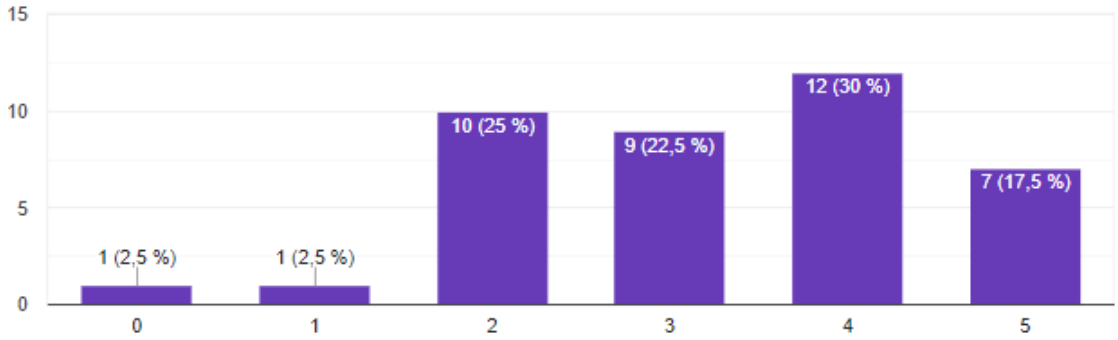
A.3.4. I want to explore and get to know the entire game with all its details. The more to discover, the better. I'm happy to take more time for that.



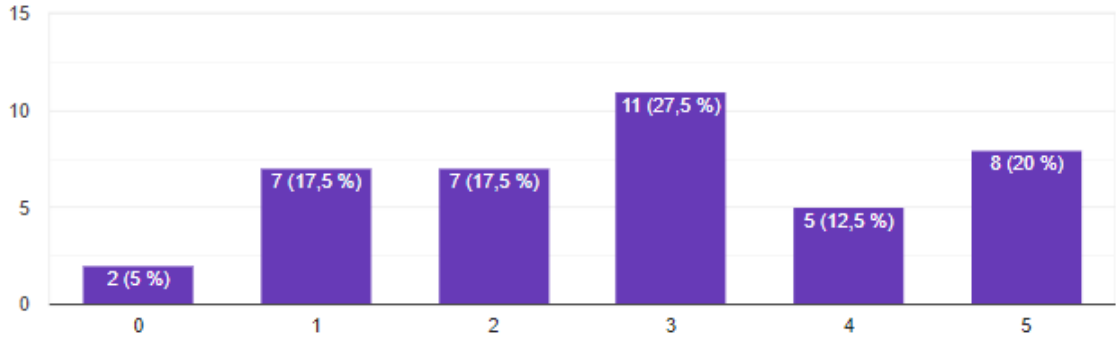
A.3.5. I want to determine the direction in which my character develops.



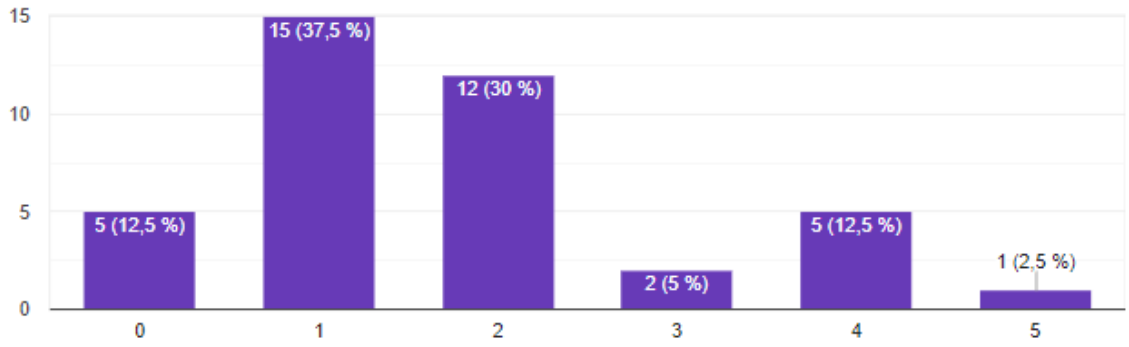
A.3.6. A game should be challenging. I want to be able to compete with other players / the AI.



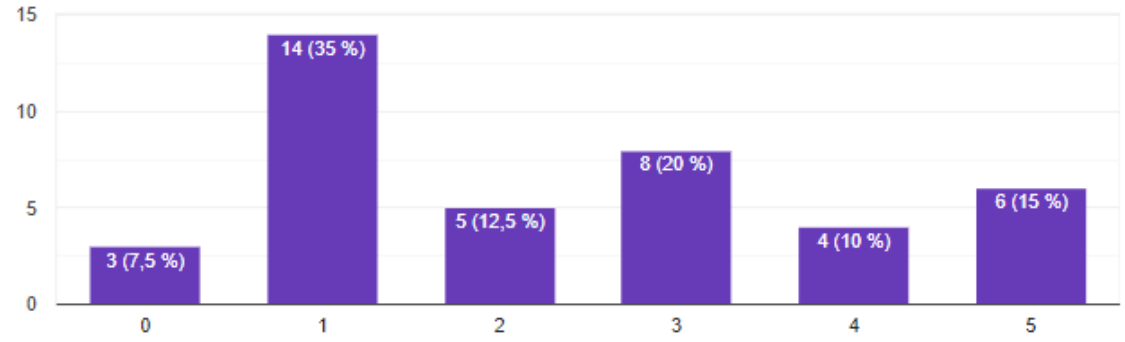
A.3.7. Without a clear goal, I have no reason to play a game. If there is no goal, I quickly lose patience and desire to play the game.



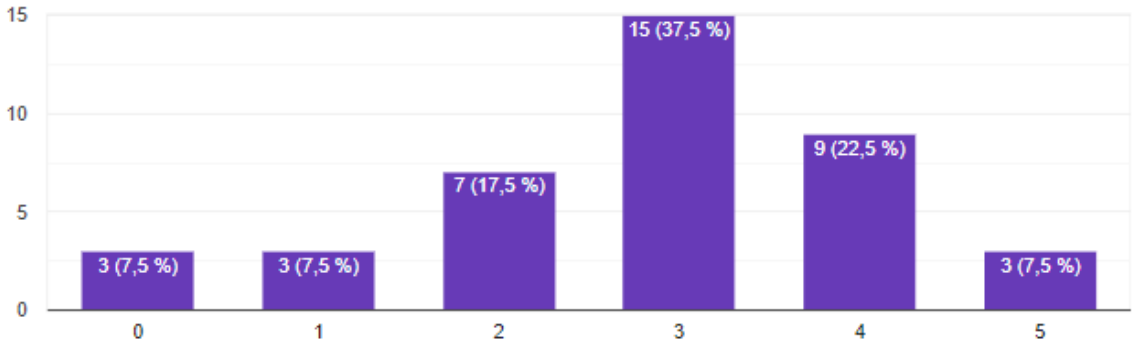
A.3.8. I want to get through the game as easily as possible. Repeating passages over and over again frustrates me so much that I tend to abandon games.



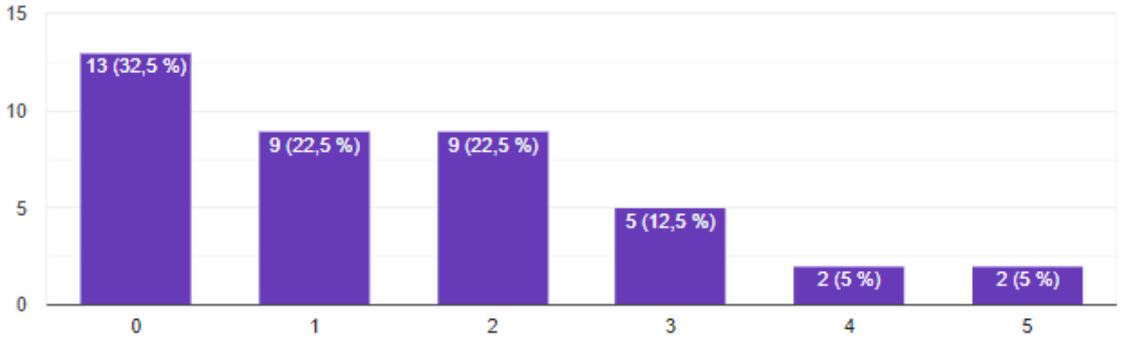
A.3.9. I prefer to play with others. Playing alone it is not so much fun.



A.3.10. I need as much variety as possible when playing. Always doing the same thing feels more like a chore than fun.



A.3.11. I prefer games with quick rounds or party games to long story modes or campaigns with lots of chapters and/or missions.



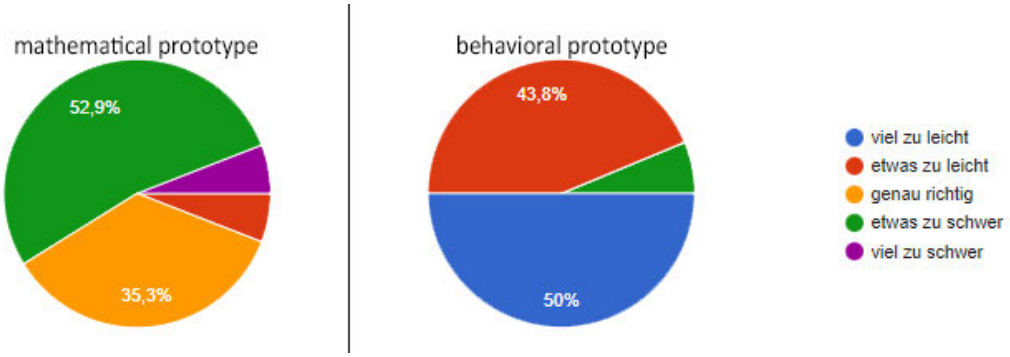
B. Prototype Test Follow-up Questionnaire

This questionnaire was filled out by the test subjects after playing one of the two prototypes (either mathematically or behaviorally balanced one). The information provided here consists of the results of the questionnaire that was taken after the initial play test. The results regarding questions about the respective prototype are separated between the answers to the mathematical approach (on the left) and the behavioral approach (on the right).
The results for the questions regarding Meta Artificial Intelligence however, are the combined answers of all participants.

A total of 33 responses were submitted, of which 17 related to the mathematical prototype and 16 to the behavioral prototype.
The questions were translated into English to keep the language of this paper consistent. Written responses, on the other hand, were not translated in order not to distort the statements and thoughts of the respective respondent.

B.1. General Questions regarding the Prototype

B.1.1. How appropriate did you perceive the overall difficulty of the prototype you tested?



B.1.2. What did you consider to be the most challenging aspect of the prototype in terms of difficulty? (also feel free to elaborate on the individual difficulty levels)

Answers for the mathematical prototype

- Die Steuerung war nicht "flüssig" genug. Ich habe vor kurzem Final Fantasy 7 Remake gespielt. Die Ausweichrolle war meiner Meinung nach zu verzögert von Druck des Knopfes bis zur Ausführung. Es fehlt eine Option zum Blocken oder gar parieren. Vorallem auf die 3. Stage bezogen ist es einfach unfair wenn dich ein Gegner mit 2 Schlägen töten kann. Du dich aber nicht heilen kannst oder mit einer Aktion (parrieren genau in der Sekunde in der er angreift z.B.) etwas dagegen tun kannst. Vielleicht wäre eine Unverwundbarkeit während der Ausweichrolle ebenfalls eine Lösung. Außerdem hat sich der Gegner nicht von den Attacken beeinflussen lassen (damit meine ich speziell eine Attacke abbrechen wenn er getroffen wird). So war es vorallem in Stage 3 fast unmöglich den Gegner anzugreifen ohne dass er dich auch trifft. Bei dem Schadensoutput macht dies das Level nicht schafftbar. (für mich)
- - Kein Abbruch (Cancel´n) der Angriffsanimation durch Ausweichen (allgemein) -Allgemein die Steuerung ist nicht Intuitiv (anvisieren auf Tab ???, Blocken nach

hinten drücken ???) - zu langsamer, träger Input (zu behebig, wenig flüssig) - fehlendes Treffer Feedback, sowohl visuel als akustisch (gefühl von Luft schlagen) - letzte Schwierigkeitsstufe unausgewogen in Hinblick auf Leben und Reaktionzeit der KI, obwohl Verhalten und Bewegungsmuster ähnlich Simpel ausfallen wie in den Ivorran gegangenen Leveln - im dritten Level wurde man zuschnell bestraft sobald man ein Angriff startet (Gefühlt Sofort Konter von der KI wenn man nur daran denkt angreifen zuwollen) - fehlendes Ausdauer Managment der KI (keine Zeitfenster um selbst zu Kontern) - kein Estus Flakon ;)

- der Sprung von 2 zu 3 war sehr extrem. 2 war first oder second try (kann mich nicht mehr ganz erinnern) 3 habe ich maximal bis etwas unter die Hälfte seines Lebens geschafft, bei vllt 10-20 Versuchen. Den Schaden des dritten Gegners habe ich als viel zu hoch empfunden. Ich war zwar durchaus ehrgeizig und wollte ihn besiegen, aber aufgrund der etwas clunky Steuerung habe ich dann die Geduld verloren. Es hat sich aber schon schaffbar angefühlt, nur war das Spiel eben nicht belohnend genug, es lange genug zu versuchen bis es klappt. Klar ist es im Rahmen der BA nicht erwartbar, dass die Mechaniken super gepolisht sind, aber für nicht-Souls-like-Spieler wie mich wären es besser gewesen, wenn der Charakter etwas schneller auf Inputs reagiert hätte).

- Die Schwierigkeit hat von Stage 2 zu 3 zu stark angezogen. Fände es da etwas smoother vielleicht besser. An sich gefällt mir dieses 2 Hit KO Thema. Hat was von den Souls spielen. Für so einen Schwierigkeitsgrad muss dann aber auch das "gameplay" stimmen. Mir ist bewusst, dass das in so einem kleinen Test nicht realisierbar ist, aber auf so einem Schwierigkeitsgrad ist sowas wie Timing (rolle kam super verspätet wenn man gedrückt hat oder garnicht) und flüssiges Spiel (Animation cancelling, hit box) wichtig. Wenn man da optimiert denke ich hat das schon richtig Potential.

- Die Unsicherheit in der Registrierung meiner Tasteneingaben kombiniert mit meiner Ungeduldigkeit. Ich muss dabei natürlich berücksichtigen, dass meine Hardware mit die Schuld dabei tragen könnte. Beim testen des 3 Stages ist mir aufgefallen das beim laufen nicht immer mein Klicken zum Angriff registriert wurde. Oder es speicherte meine mehreren panischen Klicks, dass ich in der Kombo fest steckte und nicht ausweichen konnte. Nach mehreren Versuchen in Stage 3 ist das frustrierende klicken und tippen immer zum Verhängniss geworden.

- Die Agilität: Nicht in der Lage zu sein die Ausweichrolle zu verwenden, während man sich noch in der Angriffsanimation befindet, stellte für mich die häufigste Todesursache dar. Da der Gegner mich auch treffen kann während ich mich in der ersten Angriffsanimation befinde, muss man (explizit in Stufe 3) eher raten, ob der Gegner einen Gegenangriff triggert oder nicht. Sobald man die Angriffsanimation des Gegners sieht ist man meistens bereits nicht mehr in der Lage seinen Angriff abzubrechen.
- Ich fand, das der Kampf etwas unresponsive war. Das hat sich für mich darin gezeigt, dass ein normaler Angriff eine sehr lange Animation hat und die Möglichkeit des Ausweichens langzeitig blockt (ca 3 sec). Dadurch war es sehr schwer Angriffen auszuweichen, da ich nicht auf gegnerische Angriffe reagieren konnte, sondern lernen musst, wann die KI angreifen wird, damit ich vor so einem Moment keinen eigenen Angriff starte.
- Während man selbst angreift oder nach einem Dodge kann der Gegner als Teil einer Attack-Combo (vorallem im 3. Level) viel Schaden machen, den man relativ schwer vermeiden kann. Dies motiviert auch etwas dazu, eher defensiv zu spielen.
- Die Fähigkeit "Ausweichrolle" reagierte oft zu langsam, was erfolgreiches Ausweichen erschwerte (Timing) - v.a. bei den Schwierigkeitsstufen 2 und 3, da diese Stufen durch den höheren Schaden der gegner weniger Fehler verzeihen.
- Attack Animation ging seeehr lang, Response auf Gegner Attacken war damit seehr langsam Gegner hingegen konnte sehr schnell agieren, war auch nicht stunned von treffern Außerdem sehr tanky in Level 3
- Die unpräzise Steuerung und manchmal unklaren Angriffe/Aktionen ohne Indikator. Insb. in der dritten Stufe ist dies ein Hindernis, das zu unausweichlichen Fehlversuchen führt.
- Stufe 1: Nichts Stufe 2: Die 3-Angriff-in-Folge-Combo Stufe 3: beim näheren Umrunden des Gegners "teleportierte" er sich Face Front zum Spieler und schlug ihn dadurch
- Ihm auszuweichen war bei allen Stufen schwierig, weil man nur rollen und nicht wie der Gegner seine Arme hoch machen konnte um zu blocken.
- träge Steuerung und zu viel schaden (als würde man mehrere Schwierigkeiten überspringen und von leicht, sofort zu veteran springen)

- es war nicht schwer, hat sich auf Stufe 3 'unfair' und 'nervig' angefühlt, mehr wie eine Hausaufgabe, als ein Spiel
- Stufe 3: seeeehr viel damage und sehr schwer das timing richtig zu treffen Stufe 1 und 2 eher weniger schwer
- Nur 2 Hits im maximalen Grad sind zu wenig

Answers for the behavioral prototype

- Das schwierigste war, herauszufinden, wann der Gegner angreift und wann die Rollen kommen. Man konnte recht gut vorraussehen, dass er abwehrt oder Angriffe macht, wenn man nahe stand, aber nicht, wann diese erfolgen. Ein Unterschied hierbei war der Stage 3 Bot. Nachdem ich einmal nicht direkt auf ihn losgegangen bin entwickelte er folgendes Pattern: er ist bis auf einen kurzen Abstand herangekommen. Danach ist er auf Schwertlänge an mich herangetreten, hat zugeschlagen und dann 2 Rollen von mir fortgemacht. Hier musste ich die Strategie etwas anpassen, da er sich öfter von mir fort bewegt hat und ich hinterherrennen musste.
- Stage1: Die Schwierigkeit hierbei lag meiner Meinung nach darin, die Angriffe des Gegners durch nahes heranlaufen herauszulocken und dann während der Bot nicht angreift zuzuschlagen. Stage 2: Hier kamen nun die zufälligen Bewegungen hinzu und die Angriffe waren nicht mehr so leicht hervorzulocken. Jedoch konnte man immernoch relativ gut die einzelnen Attacken hervorlocken und so den Gegner austricksen. Stage 3: Durch die hinzukommenden Dodgerolls war es nun nicht mehr möglich die Bewegungen des Gegners vorherzusagen und dies machte es extrem schwer Zeitpunkte zu finden, in denen man ungestraft angreifen kann.
- Spielstufe 3 war anspruchsvoll unter der eigenen Herrausvorderung keinen Schaden zu nehmen. Da die Hitbox des ersten Schlages sehr sehr früh einsetzt muss man hierbei sehr stark aufpassen um nicht getroffen zu werden. Das der Gegner nach ein paar schlägen direkt rollt und dem Spieler so die Möglichkeit einer Gegenattacke nimmt ist recht gut.
- Die AI innerhalb ihrer Combos zu lesen, wenn sie sich vom Spieler abwendet. Es vermittelt das Gefühl die AI folgt - anders als der Spieler - keinem Lock-On, aber versucht dennoch den Spieler zu tracken, wobei man durchaus den ein oder anderen “Seitenhieb“ einstecken muss.

- Stage 1&2: Hier war kaum ein Unterschied zu bemerken in Hinsicht auf die Schwierigkeit, daher würde ich sagen "wohin der Gegner läuft" war schwierig abzuschätzen. Stage 3: Abschätzen, wann er ausweicht und wann er angreift.
- auf allen Stufen: meine Angriffe richtig timen, sodass ich nicht selbst während meiner Attacke angegriffen werde, es ist vergleichsweise schwer Schaden zu vermeiden sobald man angreift und Spam wird mehr belohnt als Taktik
- bis auf Stufe 3 war echt ez pz, erst beim dritten bin ich mit Linksklick Spam nicht mehr weitergekommen. An dieser Stelle hat's kurz gedauert herauszufinden, wie er sich ungefähr verhält und wie ich darauf reagieren muss
- Stufe 1 empfand ich persönlich am schwierigsten, die durchgehenden angriffe gaben keine gute möglichkeit zum gegenangriff wogegen ich bei den folgenden schwierigkeitsstufen mehr das gefühl hatte
- Das Ausweichen in der letzten Stufe war besonders auffallend/schwer. Ansonsten war die größte Herausforderung die eher träge Steuerung. Die ersten zwei Stages waren sehr ähnlich.
- Dass der Charakter automatisch geblockt hat und das Angreifen unterbunden hat, auch die Range vom gegnerischen Angriff war teilweise zu groß
- Stufe1 war für mich am schwersten, durch das häufige zuschlagen des Gegners hintereinander, sodass ich nicht rechtzeitig wegrollen konnte.
- Stufe1 war für mich am schwersten, durch das häufige zuschlagen des Gegners hintereinander, sodass ich nicht rechtzeitig wegrollen konnte.
- Die Steuerung war etwas klobig und man konnte schwer die Entfernung zum Gegner einschätzen.
- Die Cooldown Zeit für Dodge Roll nach Angriff war zu hoch.
- das wegrollen des Gegners
- Dodgen zu timen

B.1.3. What did you consider to be the most easy aspect of the prototype in terms of difficulty? (also feel free to elaborate on the individual difficulty levels)

Answers for the mathematical prototype

- Die erste Stufe war recht leicht. Was nicht heißt, dass es "zu leicht" war. Es sollte in Spielen auch Gegner mit diesem Schwierigkeitsgrad geben damit Spieler einfach ein Erfolgserlebnis bekommen können. Mir fällt zwar gerade nicht ein welches Spiel es war aber ich habe mal ein Spiel gespielt indem es eine Gruppe aus Gegnern gab. Die "normalen" Gegner waren vergleichbar mit der KI von Stufe 1 und der "Boss" der Gruppe mit der KI von Stufe 2. Durch töten von Gegnern hat man sich etwas geheilt –> Einfache Gegner töten für die Heilung um es einfacher gegen den "Boss" zu haben. Für ein 1:1 war Stufe 1 definitiv zu einfach.
- Aufgrund des leichten Vorwärts-Schritts beim Angriff kann man den Gegner umkreisen und kontinuierlich angreifen und treffen, während man selbst durch diesen Schritt immer sofort wieder aus der Attack-Range des Gegners raus ist. Mit dieser Methode kann man den Gegner relativ save besiegen, ohne selbst auch nur einmal getroffen zu werden. Die "Challenge" dabei ist hauptsächlich, den perfekten Radius zu finden und einzuhalten. Anschließend wird der Kampf auch auf dem höchsten Schwierigkeitsgrad ziemlich trivial.
- Die zweite Stufe hätte mehr Anreize geben können. Da war durch umkreisen des Gegners sehr schnell die Methode gefunden den fertig zu machen. Da hätte die KI eventuell besser reagieren können? Vielleicht wenn ein Gegner zu oft das gleiche macht, dann dort entgegen wirken? Bei der dritten Stufe kam mir das so vor. Sobald ich umkreist habe, hat er immer 2 Angriffe gemacht, was mich aus dem Flow gebracht hat.
- Das blocken des Gegners in der 1 und 2 Stage ermöglichte es mehr Hits rein zu bekommen. Da es die beiden Stages sind wo ich aggressiver in meinen Angriffen war wurde er meines gefühls nach öfter in den Zustand versetzt. Ich weiß jedoch nicht ob mein rücksichtsloses angreifen das hervor gebracht hat oder die KI das random als Aktion vollzogen hat.
- - Unendliches Ausweichen und Angreifen (kein Ausdauer/Mana Managment) - Auf den ersten beiden Schwierigkeitsstufen lässt sich die KI leicht mit schnell

gelernten Bewegungsmustern (Pattern) überwinden z.B. stets nach links umkreisen - Allgemein zu wenig Variation in den Abläufen der KI

- Stufe 1: Schaden des Prototypen Stufe 2: Wartezeiten zwischen den Angriffen Stufe 3: Wartezeiten zwischen den Angriffen und, im Vergleich zum Spieler, war keine Dodge Phase möglich
- Im ersten Level hat es ausgereicht, den Gegner durch Spammen des Angriffes zu besiegen. Außerdem ist es einfach einem Angriff mithilfe einer Ausweichrolle auszuweichen.
- Die Steuerung war sehr gut und auch die Schnelligkeit der Angriffe war nach meiner Ansicht vollkommen angemessen und man konnte leicht den Gegner treffen.
- stage 1: als würde man gegen eine Puppe kämpfen, einfaches button smashen ohne viel Bewegung hat komplett ausgereicht, um den Sieg davon zu tragen
- hmm da fällt mir gerade nichts zu ein. Die ersten Gegner waren relativ leicht, weil sie verhältnismäßig wenig Schaden verursacht haben.
- Die KI ist nach etwas Spielzeit relativ gut zu durchschauen / vorhersehbar (sie rennt zum Spieler, wenn dieser zu weit weg ist).
- Die Bewegungsgeschwindigkeit des Gegners könnte etwas höher sein.
- Level 1 und 2 waren mit sehr wenigen Schlägen zu besiegen
- manchmal konnte man die Gegner stun locken
- Gegner sehr predictable
- Erlernen des Spiels

Answers for the behavioral prototype

- Die AI lässt sich auch auf Stufe 3 aufgrund eines fehlenden Lock-Ons noch relativ leicht "strafen"/umzirkeln. Wenn ab Stufe 2 die Dodge Roll dazu kommt, setz die AI diese oft noch zu Zeitpunkten ein, bei denen es recht einfach ist diesen Move zu punishen. Der Spieler kann ein Setup vollführen, indem er die dritte Instanz des Schlages vorbereitet, um den Gegner nach Ablauf der Dodge Time zu staggern. Als kleine Zusatzinformation dazu: Als ich die Stages mehrfach wiederholt habe,

habe ich diesen Ansatz getestet, da es ein ähnliches Spielgefühl zu den Soulsborne-Titeln vermittelt hat und Spieler mit Erfahrung in diesem Feld es leichter haben defensive Züge des Gegners gegen ihn zu kehren.

- Es ist schade, dass alle Gegner durch bloßes Schlagen spammen besiegt werden können. Gerade Gegner 1 schlägt nicht so häufig. solange man oft genug schlägt gewinnt man. Desweiteren können die Gegner gecheesed werden, wenn man mit gelockter Kamera einfach nur nach rechts läuft und seine Distanz über vorne und hinten leicht anpasst, kann man immer wieder aus dem Schlagtrigger Radius und der gefährlichen Distanz rein und raus tänzeln. dadurch dass man nach rechts läuft und der schlag auf der linken seite kommt trifft der Gegner kaum. Lediglich die Rollen der dritten Stufe unterbrechen diese Strategie.

- Gewinnen. Auf jeder Schwierigkeitsstufe konnte man sich einfach vor den Gegner stellen und die Angriffstaste spammen. Da der Bot zwischen Angreifen und Parieren alterniert hat (und ich nicht) starb er eher als ich. Ein Unterschied dazu machte der Stage 3 Bot, der zudem immer 2 Rollen nach einem Angriff eingeschoben hat. Aber auch hier war diese Strategie anwendbar.

- Stage 1: Es waren keine großartigen Konter vom Gegner zu erwarten, Ausweichen war ja nicht vorhanden. Stage 2: Ähnlich wie Stage 1. Stage 3: Wenn Gegner auswich durch die Rolle, konnte man ziemlich genau zu der Stelle hinrennen, wo er wieder aufstand, und damit rechnen, ihm erst einmal eine gute Kombo zu geben, da er darauf nicht direkt reagierte.

- Die gegner waren in allen schwierigkeitsstufen durch reines “draufhauen“ ohne große gefahr dabei zu sterben zu besiegen. Dabei wurde es bei stufe 1 noch am gefährlichsten, dauerte aber durch das ausweichen des gegners in stufe 3 am längsten, dafür mit der wenigsten gefahr eines gegenangriffs.

- auf allen Stufen: dem Gegner entkommen durch Ausweichen/Rolle d.h. in die anderen Ecken rennen bzw. rollen um mir sehr viel Zeit zu verschaffen; durch Angriffs-Spam gewinnen, auch wenn der Gegner zurückschlägt, richtet er insgesamt weniger Schaden an als der eigene Button-Spam

- Dur das Rollen und vermehrte Blocken des Gegners hatte man in Stufe3 gefühlt häufiger Zeit zuzuschlagen bzw. auch wieder wegzurollen, demnach hatte ich das gefühl ein defensiver Gegner, wie in Stufe3, ist leichter zu besiegen als ein offensiver, wie in Stufe1.

- Wenn man einmal die Distanz des eigenen Angriffes herausgefunden hat, so war es möglich den Gegner in die eigenen Angriffe hineinlaufen zu lassen und somit unbeschadet Angriffe durchzuführen.
- Bis auf die letzte Stufe schien einfach nur draufhauen, als Taktik zu reichen. Erst in der letzten Stufe war man gezwungen, auch selbst mal auszuweichen.
- Man konnte den Gegner leicht zwingen, die Attacke auszuführen und dann aus seiner Range entfliehen. Dadurch macht man den Gegner zum leichten Ziel.
- Linksklick Spam hat halt bis Stufe 3 als valide Option funktioniert. War auch nicht super schwer sich dem Stufe 3 Gegner anzupassen
- In Szene 3 stand der Gegner nach seinem Ausweichmanöver manchmal mit den Rücken zum Spieler und war dadurch ein leichteres Ziel.
- Sehr viel eigene HP, Gegner dodged in falschen Momenten, Attacken schnell nacheinander zu spamen half oft den Gegner zu töten
- Den Gegner "auszumanövern" indem man die Reichweite des Schwertes ausnutzt, um den Gegner zu pieksen und dabei zu umkreisen.
- Den Kampf, es war recht einfach zu gewinnen ohne viel Ausweichen oder sich bewegen zu müssen
- Steuerung

B.1.4. What would you change to adjust the difficulty level of the prototype to feel "just right"? (also feel free to describe your approach to each difficulty level).

Answers for the mathematical prototype

- Ich denke er ist grundsätzlich schon ganz gut. Eventuell eine Stufe mehr rein bringen um die Schwierigkeitsteigerung besser anzupassen. Ansonsten wenn es mal Richtung richtiges Spiel geht, ein flüssigeres Spiel ermöglichen. Dann kann man sich auch an den härteren Gegnern versuchen ohne Frust zu haben nach dem Motto "ich hab doch ausweichen gedrückt... er hats nicht gemacht.."

- den Schaden der 3. Stufe senken, aber dafür ihr Leben erhöhen. Dadurch wäre der Gegner weniger demotivierend bzw. man würde mehr experimentieren, wie an seiner Verteidigung vorbei kommt ohne einen Schlag zu kassieren, statt den vollen Fokus darauf zu legen, nicht getroffen zu werden.
- weniger Schaden, schnellere Steuerung stage 1: einfach nur button smashen stage 2: viel in Bewegung geblieben und versucht, auf Distanz zu bleiben stage 3: keine Chance das manuelle parieren hat mir sehr gefehlt (ich weis, ich hab zu hohe Ansprüche)
- Höhere Mobility des Gegners auf allen Stufen (das er Dodgen kann zB.) und kein automatischs Blocken - dies könnte mit Rechtsklick erfolgen. Allg. sollte der Gegner nicht so "Clunky" sein. Damage Werte an sich passen für diese 3-Level-Präsentation
- "snappier" movement. Zur Zeit empfinde ich das movement/attack system als zu unresponsive bzw zu langsam da man nicht aus einem schwerthieb raus dodgen kann, jedoch eine Attacke lange braucht um zu connecten
- Ich glaube das habe ich schon recht gut bei der Frage nach zu schwer beantwortet. Blocken/Parieren einführen, Ausweichrolle intuitiver/flüssiger machen, Gegner Angriff abrechen lassen wenn er getroffen wird.
- Vielleicht die Attack-Animation überarbeiten oder die Attack-Range des Gegners anpassen, sodass die oben beschriebene cheesy strategy entweder nicht mehr funktioniert oder sehr viel schwieriger wird.
- Um den Spielspaß über einen längeren Zeitraum zu gewähren, sollte eine Balance aus Modifikatoren, wie Leben und genommener Schaden, sowie Fähigkeiten der AI gefunden werden.
- - Bessere Steuerung - schnellerer Input bei Tasteneingaben - 360Crad Roll vermögen (fühlt sich wie 8 Wege System an) - "Cancel´n" von Angriffen mittels Rollen
- Auf jeden Fall die Reaktionsgeschwindigkeit für die Ausweichrolle verbessern, sodass diese im Kampf nützlich wird.
- Animations/Action cancelling würde ungemein helfen. Vor allem in Stage 3 wo eine zwei Hit-Combo dich erlegt.
- Ich fand die Schwierigkeit im 2. Level schon sehr nah, damit es sich richtig anfühlt.

- Mehr hit punkte für 1 und 2 Schnellere Reaktionsmöglichkeit mit der Rolle
- Die Menge an HP für mich und den Gegner sind gut aufeinander abgestimmt.
- Ich fand die Schwierigkeitsstufen vollkommen in Ordnung.
- War für mich sehr passend :D
- 3-4 Hits auf Stufe 3

Answers for the behavioral prototype

- Ich gehe hier nach Features vor, die ich ab einer Stufe einführen und dann konsequent beibehalten oder ausbauen würde. Stufe 1: Ab hier sollte die AI in einem 1v1 Szenario einen Lock-On, gleichsam dem des Spielers verwenden. Es erlaubt dem Spieler mehr Klarheit beim Lesen der Bewegungen und lässt die AI zielgerichteter wirken/agieren. Stufe 2: Die AI sollte Dodge Rolls sparsamer einsetzen. Oft führt die AI zwei Dodge Rolls hintereinander aus und das resultiert darin, dass die AI sich gern auch mal zu nah an eine Wand begibt und offener für Combos ist, aufgrund des begrenzten Bewegungsspielraums. Stufe 3: Hier könnten noch weitere Kniffe dazu kommen. Z.B. könnte die AI ohne Lock-On nach einem verfehlten Schlag Combos abbrechen und sich repositionieren/dodgen, statt weitere Hiebe in die Leere auszuführen. Weiterhin könnte sie während den Angriffen auch weiterhin Möglichkeiten der Bewegung nutzen oder auch den Spieler für unüberlegte Dodge Rolls in ihrer Angriffsreichweite bestrafen.
- Mein erfolgsversprechendstes Vorgehen bei Stufe1 war stupide auf den Gegner einzuschlagen, da ich das Gefühl hatte, wegrollen lohnt sich nicht. Auch, weil man nicht die Möglichkeit hat, während des eigenen Schlagens wegzurollen, bzw. wenn man während des Schlagens wegrollen möchte, auch nachdem der Schlag zuende ist, nichts passiert. Bei Stufe2 war es ähnlich. Bei Stufe3 hat es sich schon ab und zu gelohnt zu rollen, Haupttaktik war jedoch auch hier ehr nur draufschlagen. Die KI selber erschien mir, als würde sie auf meine Angriffe “logisch“ reagieren, doch dadurch, dass sie nicht alles dogen kann und auch beim Blocken Schaden bekommt, war Draufschlagen meist erfolgsversprechend. Vielleicht könnte man eine Mechanik einbauen, sodass man, falls der eigene Schlag geblockt wird, mit einer gewissen Wahrscheinlichkeit kurze Zeit “gestunned“ ist. Dafür sollte das Rollen/Ausweichen mehr Reward bringen.

- Die Angriffe der Bots müssten konsequenter sein, also mehr Schaden verursachen. Somit wäre es um einiges gefährlicher, sich dem Bot für einen Schlag zu nähern, und man müsste tatsächlich genauer auf das Verhalten des Bots achten, um Fenster für Angriffe zu finden. Das Powerscaling müsste auch um einiges straffer sein, sodass Stage 2 gerade so noch machbar ist, und Stage 3 wirklich Zeit und Anstrengung benötigt, um gemeistert zu werden. Und somit eben eine Herausforderung ist. Dem Stage 3 Bot könnten zudem Fähigkeiten gegeben werden, die der Spieler nicht hat, und um die herum er arbeiten muss - und immer neue Wege finden muss, da ja die KI die Strategie des Spielers irgendwann kontert. Die Kämpfe sollten allgemein länger gehen, da in dieser kurzen Zeit (pro Stage vllt 1 Minute) der Effekt eines "lernenden" Gegners quasi nicht spürbar war.

- Das Zeitfenster für den eigenen Angriff muss etwas deutlicher sein sodass gutes Timing belohnt wird, denn Taktik und gezielte Angriffe sind sehr schwierig (Frage 1 begründet). Dafür muss aber auch außerhalb dieses Zeitfensters der Gegner gefährlicher sein (blocken, parieren, härter zurückschlagen, etc.), denn im Moment gewinnt man jeden Schlagabtausch durch Spam sobald man mit der eigenen Richtung umgehen kann und nicht daneben schlägt. Das liegt wie in der letzten Frage beschrieben daran, dass man als Spieler auch im Schlagabtausch eigentlich fast immer gewinnt. Es ist relativ einfach auch auf höheren Stufen durch aggressiven Spielstil zu gewinnen, da der Gegner eher dazu tendiert mehr auszuweichen und nicht mehr Schaden anzurichten.

- Der Gegner Stufe 3 ist definitiv der Gegner welcher den meißten spielspaß liefert. Die Steigerung der Komplexität ergibt definitiv Sinn, jedoch ist es zu einfach angriffe zu spammen. Der Schaden der einzelnen Gegner sollte definitiv erhöht werden, sodas sie auch gewinnen können obwohl sie weniger angreifen. Desweiteren könnte mit der länge des Hitstuns experimentiert werden. Einfache Gegner werden eher gestunnt durch angriffe und hohlen länger aus um ihren Angriff durch zu führen. Schwerere Gegner sind schwerer zu erwischen, da sie weniger Angriffsmomente bieten. desweiteren könnte man für Stufe 3 einfach 2 Gegner nehmen, durch das aufpassen auf 2 Gegner würde die Komplexität und der Schwierigkeitsgrad ordentlich steigen.

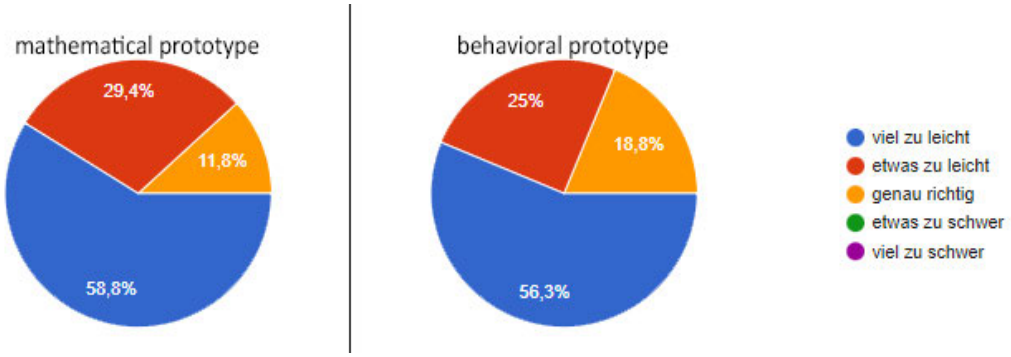
- Eine Anpassung bei den ersten beiden Schwierigkeitsstufen ist es meiner Meinung nach nicht nötig. Bei der dritten Schwierigkeitsstufe hatte ich teilweise das Gefühl, dass es sich eher zufällig anfühlte wenn ich den Gegner treffen konnte und nicht

als hätte ich etwas besonders gut gemacht. Man könnte dies vielleicht anpassen, indem man den Gegner auch nur dann dodgen lässt, wenn der Spieler auch wirklich angreift. Desweiteren wäre vielleicht ein kleiner Cooldown auf dem dodge vorteilhaft, um dem Spieler auch Zeitfenster zu geben die er sich quasi erarbeiten kann.

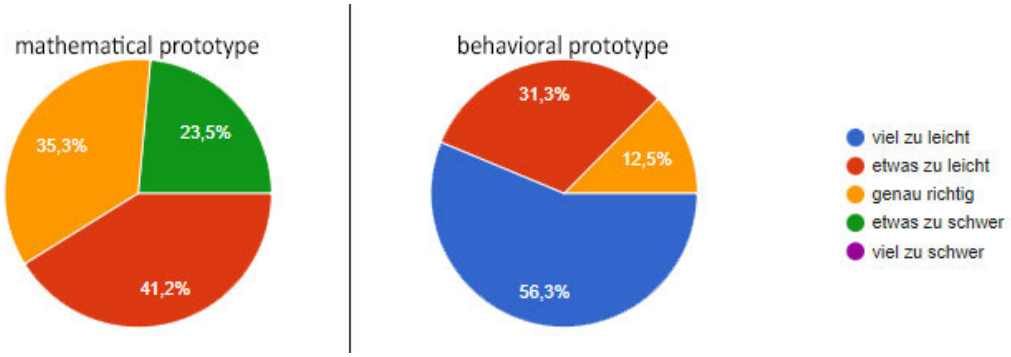
- Ich würde entweder die maximalen HP des spielers verringern oder den schaden des gegners erhöhen. Damit würde es notwendiger werden, die spielmechanik des ausweichens und des blockens zu verwenden. Dafür könnte man dann die zeit zwischen den angriffen des gegners auf der einfachsten schwierigkeitsstufe verlängern und diese zeit bei steigender schwierigkeit kürzen. Das ließe im einfachsten fall mehr zeit zum ausweichen.
- Bei Stufe 1 und 2 würde ich den Spieler etwas mehr für Fehltritte bestrafen, also mehr Hp des Gegner, womit ein Gegenschlag anteilmäßig stärker ist als der eigene, oder so. Außerdem noch minimal komplexeres Verhalten, dass man nicht einfach nur stumpf draufhauen kann. Stufe 3 sollte vielleicht etwas transparenter und nich (gefühlt) gaaanz so random sein
- Meiner Meinung nach weicht mir der Stage-3-Gegner zu oft aus ins nichts, anstatt effektiv die Angriffe auszuweichen, was aber möglicherweise gar nicht so einfach machbar ist. Allgemein greifen die Gegner nur einmal an, anstatt hin und wieder die "3er-Combo" durchzuführen.
- Der Gegner sollte etwas aggressivere vorgehen und in richtigen Momenten seine Defensive vernachlässigen, um öfter anzugreifen. Z.B.: Wenn der Spiele gerade ein paar mal getroffen wurde und deswegen die Flucht ergreifen will, wäre das eine ideale Situation zum Angreifen.
- Unterschiedliche Angriffsgeschwindigkeiten für unterschiedlich starke Angriffe, Panzerung, die man erst überwinden muss und ein wenig mehr Intelligenz beim Gegner Auch das Rückwärtsausweichen wäre ein schönes Feature
- Ihn evtl. auf verschiedene Angriffsmuster des Spielers reagieren lassen. Z.B. wenn er eine 3er-Kombo ansetzt, direkt eine Kombo zurückgeben, da es ja (meines Erachtens nach) einen kurzen Cooldown gibt.

- Besonders das movement des NPCs war eher einfach gestaltet und könnte deutlich mehr Schwierigkeit hinzufügen. Auch das Ausweichen hatte auf mich eher einen zufälligen als sinnvollen Charakter.
- schnellere und gezieltere Angriffe der KI, besseres timen und bessere RIchtungswahl des dodgens
- Reaktionsgeschwindigkeit und Anpassungsfähigkeit an gegnerische Attacken ändern
- Der Gegner sollte zwischendurch abwartender sein, um den Spieler zu locken.

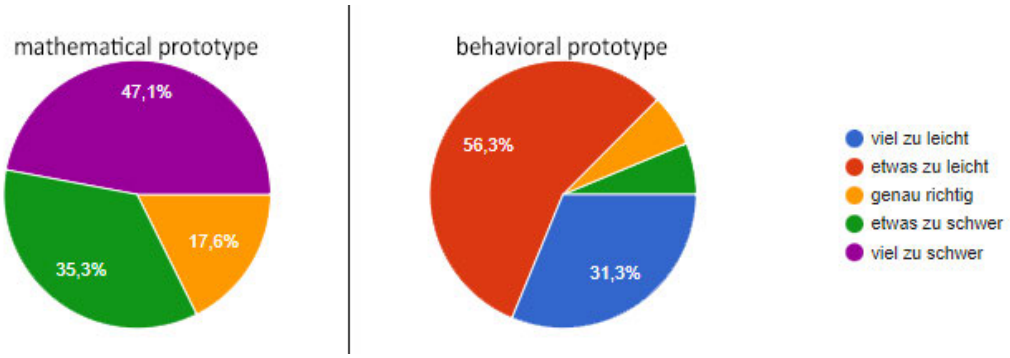
B.1.5. How high did you perceive the difficulty of the 1st stage of the prototype you tested?



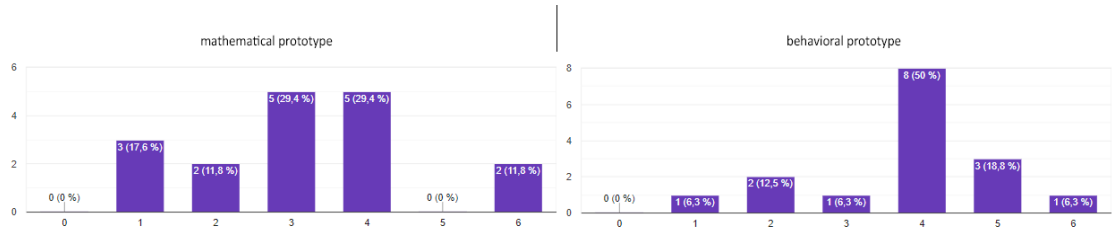
B.1.6. How high did you perceive the difficulty of the 2nd stage of the prototype you tested?



B.1.7. How high did you perceive the difficulty of the 3rd stage of the prototype you tested?



B.1.8. How well did you get along with the controls of the player character?



B.1.9. Is there anything you would want to improve about the controls and handling?

Answers for the mathematical prototype

- - Schnellerer Input - 360Crad Ausweichrolle - Abbuch (Cancel) der Angriffsanimation mittels Ausweichen oder Blocken - nach einem (eigenen) Schlag zu große Verzögerung zur Roll Animation - schnelle und Starke (langsame) Angriffe - Eigene Block Taste - Treffer Feedback (visuel und akustisch) - Anvisieren auf andere Taste (unbedingt) gern Controllerunterstützung gesehen - Anvisieren des Gegners , auch wenn er nicht direkt im Kamera Blickwinkel steht - Pototyp Für Dolch und Hammer (diversere Möglichkeiten in Angriffsgeschwindigkeit und Schadens output (auch

für KI)) - Wo ist das Parieren ? Wenn der Protagonist rollt, hebt er am Ende der Animation leicht ab. Nicht schlimm, aber leicht Irritierend

- Er hat sich zu starr angefühlt. Der Übergang von Animationen (und der damit verbundenen Funktion) hat zu lange gedauert. Kann natürlich auch sein, dass es aktuell einen Cooldown gibt zwischen den Aktionen. Die Steuerung hat sich sehr eingeschränkt angefüllt wenn der "Zielmodus" aktiviert war. Mir ist bewusst, dass sich die Figur dann relativ zum Ziel bewegt. Aber gefühlt hat sich meine Figur dann auch langsamer bewegt. Das hat den Spielfluss gestört
- Ja, schnellere Reaktionszeiten auf Input. Der Sprung hat sich auch etwas clunky angefühlt, irgendwie unrund im letzen Teil der Animation. Diese würde ich mir auch noch einmal anschauen. Und das Parieren würde ich als extra Taste belegen (maybe Rechtsklick), da die Sprung-Taste auch exklusiv für die Ausweichrolle da sein sollte.
- Dodge fühlt sich etwas clunky an, könnte aber auch an der Animation liegen. Eventuell könnte auch dezentes Input-Queueing helfen, sodass man nach einem Angriff/Dodge ohne Delay smooth erneut dodgen kann. Das macht es aber natürlich auch leichter.
- Dodge Rolle muss sofort ausführbar sein (unterbricht auch den eignen Angriff um gegnerischen Angriff zu dodgen). kein Automatisches Blocken - stattdessen Rightclick to Block
- Fühlte sich sehr unsmooth an, Kamera Bewegung mit der Maus auch zu langsam imo, vlt sowas wie blocken oder Counter einfügen
- siehe frühere Antwort; Der Spielercharakter muss sich schneller bewegen können/mehr unter der Kontrolle des Spielers stehen
- Wenn ich Rolle drücke sollte diese auch sofort kommen. Wenn dafür auch die Angriffsanimation unterbrochen wird ist das gut.
- Da ich den Prototyp auf meinen Laptop teste, wäre eine Option den Angriff auf E oder eine andere Taste zu setzen hilfreich.
- Ich finde, dass die Angriffe die Möglichkeit Ausweichrollen auszuführen zu lange blockiert hat.

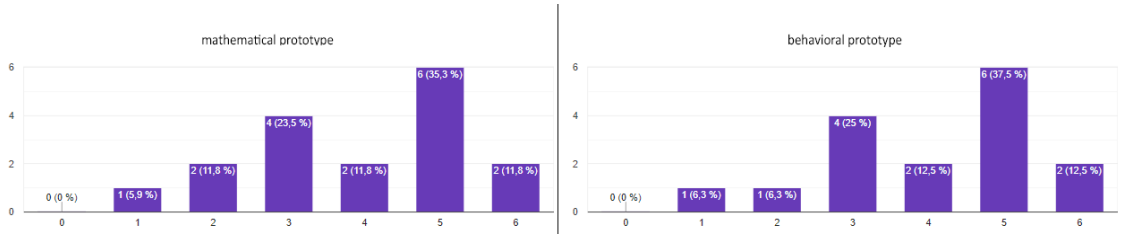
- Ausweichen hatte gefühlt etwas "Input-Lag" (teilweise erst bei zweitem Drücken ausgeführt)
- Rolle immer ausführbar nicht geblockt von Attack Animation bzw Attack cancelbar
- Weniger träge Animationen, Die Möglichkeit Aktionen abzubrechen.
- Die Ausweichrolle sollte aus "AnyState" heraus triggerbar sein.
- Das genauere und schnellere Auslösen der Ausweichrolle.
- schnellere Steuerung, manuelles parieren
- nein

Answers for the behavioral prototype

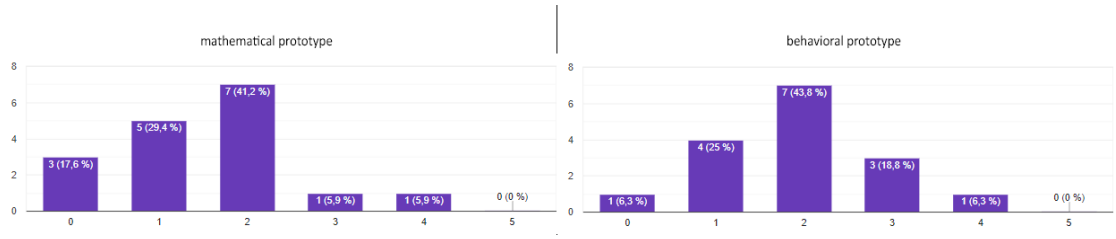
- Der kleine "Bump", wenn ein Character aus der Dodge Roll wieder auf die Füße kommt war im ersten Moment etwas irritierend. Der Stagger im Hit Feedback passiert etwas zu "ruckartig" und endet vor allem auch in einer solchen Weise. Eine kurze Recovery Animation könnte helfen. Z.B.: Character taumelt etwas zurück, richtet sich wieder zur Idle Haltung auf und ist dann auch wieder kontrollierbar.
- Es könnte etwas schneller sein. Angriffs -und Ausweichanimationen sollte man unterbrechen können mit Bewegung oder anderen Aktionen. Jetzt muss man immer warten, bis der Angriff durchgeführt wurde, anstatt spontan auszuweichen und den Angriff zu unterbrechen.
- Die Rolle hat einen kleinen Hacken, wodurch sie sich komisch zu verwenden anfühlt, desweiteren wäre ein auto aim bei nicht gelockter Cam sinnvoll, da man so sonnst oft einfach nur ins leere schlägt. Ich habe das spiel eigentlich nicht ohne auto aim gespielt
- Die Rolle fühlte sich sehr undynamisch an. Oftmals konnte man nicht rollen, weil man noch in den letzten Animations-frames eines Angriffs, eines Schrittes oder einer anderen Rolle war. Dadurch hat sich die Bewegung sehr eingeschränkt angefühlt.
- Es fühlte sich teils etwas sehr undynamisch an, dass man immer auf den kompletten Angriff warten musste und nicht währendessen dodgen konnte. Dies wirkte sich für mich auch negativ auf das Schwierigkeitsempfinden aus.

- Dieses "Sliden" der Kamera finde ich persönlich nicht so schön. Auch hatte ich das Gefühl, dass manche Aktionen delayed waren wie in etwa das Ausweichen.
- Sehr träge Steuerung. Die Blocken-Funktion war nicht so recht intuitiv/hat sich zufällig angefühlt. Das Ausweichen hat gefühlt zu lange gedauert.
- Belegung der Fokustaste auf den Gegner abändern, man musste den finger von "A" dazu benutzen, dadurch in eine richtung "bewegungsunfähig"
- Ich hätte gerne eine einfache Möglichkeit meine Blickrichtung während des Laufens zu ändern ohne meine Laufrichtung zu ändern.
- Rückwärtsausweichen bitte hinzufügen, unterschiedlich starke Angriffe, Regenerationsbonus wie beim Gegner
- Meckern auf hohem niveau... aber ein slider für die mausempfindlichkeit wäre cool gewesen.
- Maybe bei Freikamera nich ganz so schwammig, aber mit Enemy Focus war eigentlich tight
- Mausrotation fühlt sich schwammig und träge an
- Die Reaktionszeit sollte verkürzt werden.
- Frei wählbare Tastenbelegung
- nein

B.1.10. How pleasing did you find the graphic appearance?



B.1.11. How did you feel about the game mechanics provided? (Character movement, combat system)



B.1.12. How would you customize the combat and prototypes to create a better and more enjoyable gaming experience for you?

Answers for the mathematical prototype

- Blocken/Parieren einbauen, Ausweichrolle verbessern, Gegner bricht Attacke ab bei Treffer, Animationen flüssiger werden lassen. Entweder passive Heilung (kurz wegrennen und dann regeneriert sich das Leben) oder eine aktive Heilung (durch töten von "einfachen" gegnern oder halt items) einbauen. Das ist aber natürlich sehr subjektiv. Ich mag es nicht "unbesiegbare" Bosse zu haben. Lieber habe ich herausfordernde Bosse die ich entweder mit viel Skill (punktgenau parieren wie z.B. die Wächter in The Legend of Zelda BOTW) oder eben durch das gute alte "Schaden machen, kurz in Deckung heilen" (so ziemlich jeder moderne Ego Shooter mit passiver Heilung oder auch sowas wie Horizon Forbidden West mit aktiver Heilung) besiegen kann. Deswegen habe ich auch Elden Ring nach 30 Minuten wieder deinstalliert. Es ist mir zu herausfordern für den Spaß den es mit dem Kampfsystem bietet. In Horizon Forbidden West gibt es ebenfalls sehr herausfordernde Kämpfe aber das sehr gute Kampfsystem machen es eben "fairer" für den Spieler. Keine Ahnung ob das hier alles Sinn ergibt :D
- Erstmal das Handling, wie oben schon beschrieben. Gameplay-technisch vielleicht in eine Art Schere-Stein-Papier System umwandeln: Blocken schlägt Angreifen (man reduziert den Schaden und betäubt den Gegner wenn er genau zu dem Zeitpunkt angreift und hat somit Zeit für einen Gegenschlag), Angreifen schlägt Ausweichen (die Rolle ist schneller als ein Block, aber langsam genug, dass man

mit einer Angriffscombo hinterherkommt und trifft) und Ausweichen schlägt Block (die Blockanimation ist so lang, dass man, wenn der Gegner nicht angreift selber einen Konter riskiert (ähnlich wie der Counter aus Smash Bros.).

- Ich würde ermöglichen früher nach einem Angriff eine Ausweichrolle auszuführen und potenziell auch die Möglichkeit einbauen, einen Angriff bis zu einem gewissen Zeitpunkt mit einer Ausweichrolle abzubrechen. Alternativ könnte das Poise System ausgebaut werden. Jedenfalls erschien es so, als würde die KI teilweise Angriffe abbrechen, wenn sie rechtzeitig getroffen werden. Dann nahm sie eine Verteidigungshaltung ein. Ich weiß aber nicht, wann so etwas genau passiert ist und wie ich es ausnutzen kann.

- - mehr Verhaltensmuster - Ausdauerleiste für Konter Zeitfenster, als auch eigenes Angriffs und Ausweich Managment - schnelle und langsame (starke) Schläge für mehr Varianz - Im letzten Level : die KI setzt ein Schlag, dreht sich um 180Crad (wenn umkreist) und schlägt ein weiteres Mal zu, mitten in der Dreh Animation (falls man in Reichweite ist). erzeugt Frust weil kaum möglich ist darauf zu reagieren - Treffer feedback (visuel und akustisch) - wo ist das Parieren ?

- Eine etwas abgetöntere Umgebung, damit diese für die Augen nicht zu anstrengend ist. Außerdem wäre es gut, nachvollziehen zu können, wann sich die Charaktere mit beiden Armen schützen - in dieser Zeit könnte etwa der Spieler, wenn der Gegner gerade beide Arme nach oben vor das Gesicht hält, noch einmal zuschlagen, was mehr Herausforderung in Sachen Reaktionsfähigkeit und Koordination mit sich bringen würde.

- ich würde die verschiedenen Gegner anders groß machen (1 Stufe kleiner als, 3 Stufe größer als der Spieler). Die Gegner geben schon zu sehen, dass sie angreifen, man kann das nur schlecht ausnutzen, da man für die eigene Dodge Roll zu lange warten muss nach einem Angriff

- Ich hätte gern leichte und schwere Hiebe gehabt. Die unterschiedlich lange dauern und entsprechen unterschiedlich Schaden machen. Ansonsten die genannten Themen wie, dass man Animationen abbrechen kann und die Steuerung genauer und sofort reagiert.

- Eventuell komplexere Machaniken hinzufügen, die den Spieler mehr für gutes Timing/Dodgen/etc belohnen, z.B. Stamina-System oder Cooldowns/Stun. Eine weitere Option wären defensive "Timing-Skillchecks" wie Parry/Blocken.

- Die Ausweichmanöver beim Angriff des Gegners finde ich ein wenig zu langsam, jedoch erhöht es den Schwierigkeitsgrad und je nachdem kann dies auch beibehalten werden, falls dies gewünscht sei.
- Möglichkeit zu Blocken /Kontern wünschenswert, Beim Ausweichen kommt zum am Ende der Rolle eine leichte "Hüpfbewegung", Animation sieht dort etwas holprig aus
- Die Möglichkeit seine eigenen Angriffe abzubrechen wäre sehr angenehm. Eventuell könnte auch eine Block- oder Parry-Mechanik Abhilfe schaffen.
- Allg. "smoother" Bewegungsabläufe und Animationen, sowohl beim Gegner als auch beim Spieler. Gefixte Hitboxen :3
- Weniger träge Animationen, sauberere Steuerung, Gegnerische AI nicht automatisch mit "Aimbot" ausstatten.
- manuelles parieren, beim parieren weniger Schaden erleiden
- Blocken/Counter, Maus Geschwindigkeit anpassen
- z.B. Option für Parry von Attacken
- Animation/Action cancelling

Answers for the behavioral prototype

- Ich versuche hier ein paar Feature-Vorschläge zu bringen mit Bedacht auf deren potenziellen Umfang im Einbau: 1. Eine zweite Art des Angriffs (Heavy Attack) mit einer eigenen Combo-Linie. Das würde eine größere Variety an möglichen Moves mit unterschiedlichen Timings bieten. 2. Combos, die sich aus einer Light/Heavy Attack Abfolge mischen lassen, ähnlich wie in Beat'em Ups/Brawlern -> Light + Heavy + Light // Heavy + Light+ Light. 3. Stagger/Knockback Time/Intensity abhängig davon, von welchem Move ein Character getroffen wurde. 4. Ein Ressource wie Stamina, die vom Spieler mit Bedacht gemanaged werden muss und so Buttonmashing riskanter macht, egal ob bei offensiven oder defensiven Aktionen. 5. Da wir uns hier in einem Schwertkampf Prototypen befinden: Eine Möglichkeit der Parade, die eine Riposte/Counter Aktion triggern kann und so dem in die Ecke gedrängten Kämpfer ein Comeback ermöglicht und den Angreifer genauer abwägen lässt, wie lange ein Ziel vermutlich noch im Stagger Status verbleiben wird.

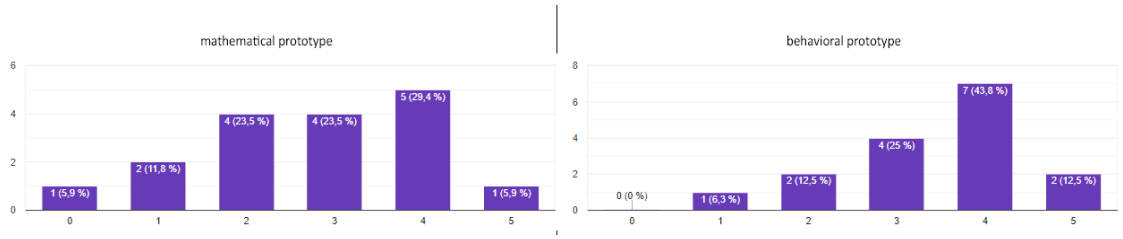
- Ich bin ein großer Fan von Animation-Canceling und schnellen Vorgängen. Die langsamen, schweren und den Charakter in der Position festhaltenden Angriffe haben sich für mich sehr quälend angefühlt. Man könnte dies damit kompensieren, dass sich die Angriffe sehr mächtig anfühlen, und viel Schaden verursachen. Oder man erhöht die Geschwindigkeit der Animation, erlaubt schnelle Wechsel von Angriffen, Verteidigungsschlägen und Rollen. Eine Abwehr-Mechanik würde dem ganzen vielleicht noch mehr taktische Tiefe verleihen.
- Das rollen sollte meiner meinung nach einen laufenden angriff unterbrechen können. Das würde das ausweichen sehr viel reaktiver machen. Meistens konnte man einem gegenangriff gar nicht mehr ausweichen weil dieser schon getroffen hatte bevor man mit dem rollen starten konnte.
- unterschiedlich starke Angriffe und damit auch unterschiedlich schnelle Angriffe Darüber hinaus: das Tutorial kann man nicht beenden/verlassen. Man lernt die Steuerung kennen, darf dann im nächsten Raum einen Testgegner bezwingen und dann bleibt man stecken im Tutorial
- der Gegner sollte in der Lage sein gegen Spam von Angriffen zu reagieren (z.B. durch blocken, parieren, kritische Gegentreffer, etc.); es ist durch Rollen zu einfach Distanz zu gewinnen, evtl. den Spieler mehr limitieren oder den Gegner befähigen besser zu folgen
- Blocken als eigene Aktion, z.b. auf Rechtsklick. Ausweichen als verschiedene Bewegungen z.B. Ausfallschritt/rechts-links ducken/vorwärtsrolle. Belohnung für combos (mehr schaden, wenn 2-3 fach combo ausgeführt etc)
- Ich würde wahrscheinlich nur das attack-cancelling hinzufügen, damit man einfacher auf die Angriffe des Gegners reagieren kann. An sich wäre Blocken auch ganz nett, ist aber kein muss.
- Die Combos waren Cool, aber es wäre wunderschön, könnte man auch gezielt blocken mit Rechtsklick, das würde auch Gegner komplizierter machen.
- bisschen mehr Möglichkeiten im Kampf, so Kombos mit Sprung, Parry und so. Dann eventuell noch bisschen Environment (Bäume, Fässer oder so)
- Rollen auch während man einen Angriff gestartet hat (Angriff dafür abbrechen bzw. priorisieren). Möglichkeit selbst zu blocken.

- Ich persönlich mag so eine Art von Spielen nicht besonders, daher wäre mein Feedback hier nicht spezifisch genug.
- Eine Erklärung und bessere Resonanz wann und wie man blockt und warum man das tun sollte.
- Alles etwas schneller und ein etwas aggressiverer Gegner in hören Schwierigkeitsgraden.
- Bessere Grafik, mehr Angriffsmodi, schwierigere KI
- Ein aktives Blocking hinzufügen.
- Attackenvariation

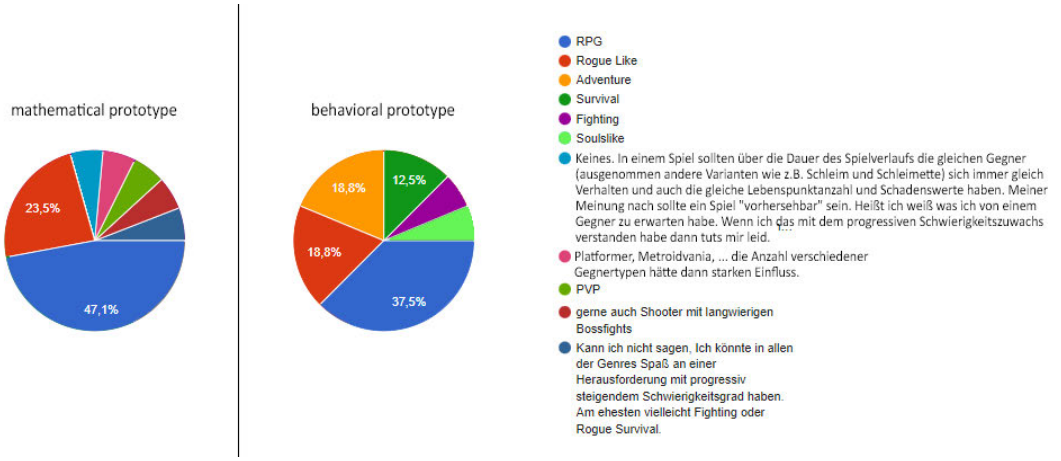
B.2. Evaluation of Gaming Behavior with regard to Game Duration and Replay Value

Next, participants were asked to imagine the previously played prototype as a finished game in which the progressive difficulty increases were modeled on the difficulty levels of the prototype.

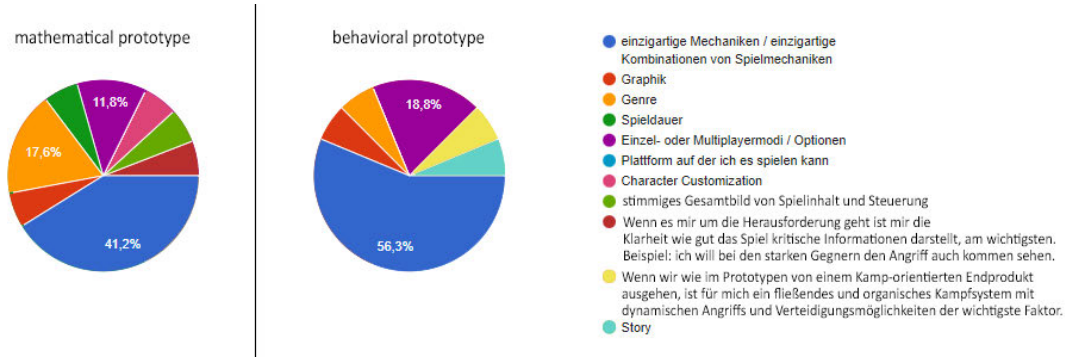
B.2.1. How likely would you be to buy such a game?



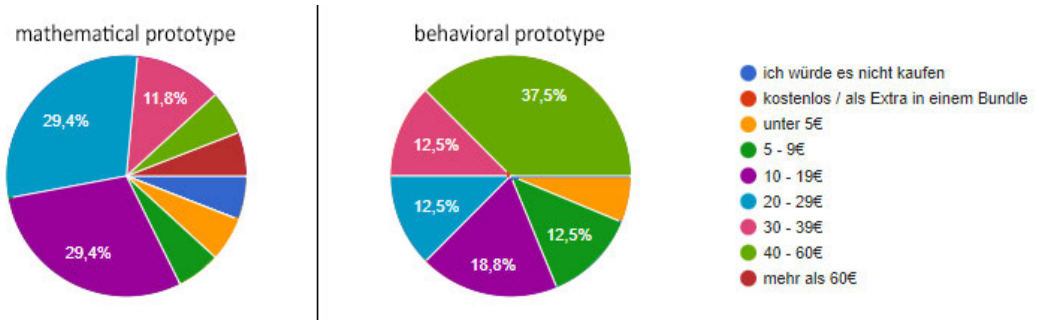
B.2.2. What genre would the game need to possess to make it attractive for you to buy?



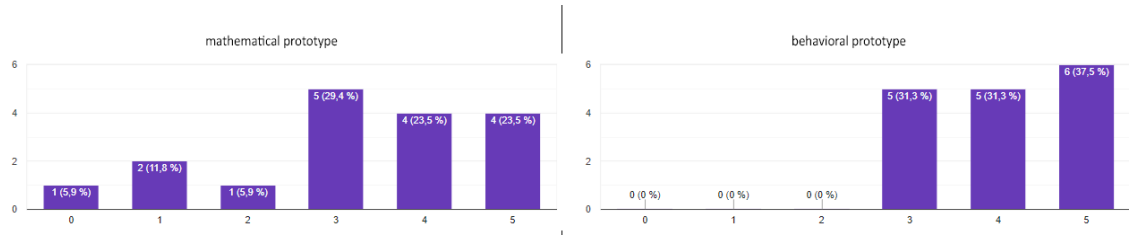
B.2.3. Which criteria would be the most decisive for you when buying such a game?



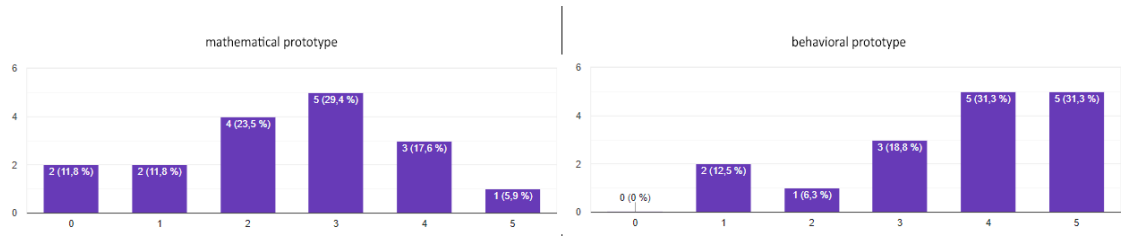
B.2.4. What is the maximum amount of money you would be willing to spend on this game?



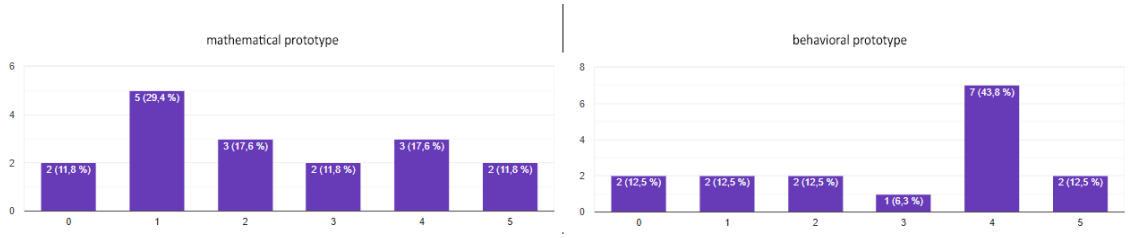
B.2.5. If you already owned the game, how likely would you be to play through it completely?



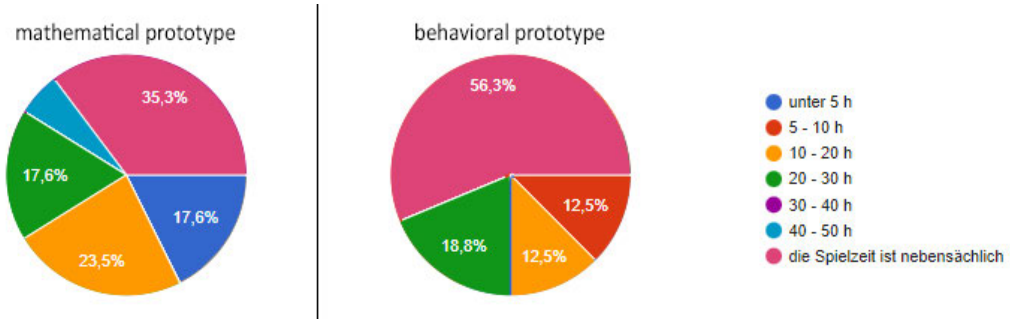
B.2.6. How likely would you be to invest time in the game after finishing the "normal mode"/story for extra content like optional bosses or dungeons?



B.2.7. How likely would you be to replay the game purely on the basis of difficulty progression if the rest of the game (mechanics, graphics, story, etc.) appealed to you?



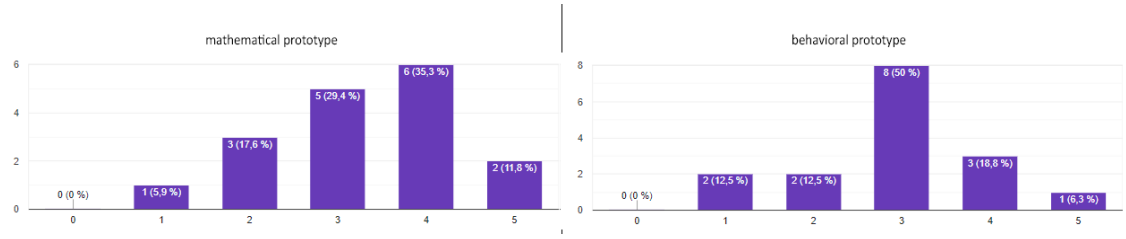
B.2.8. In regards to replaying the game, what should be the maximum gameplay time required for a “story mode“ to get you more excited to play again?



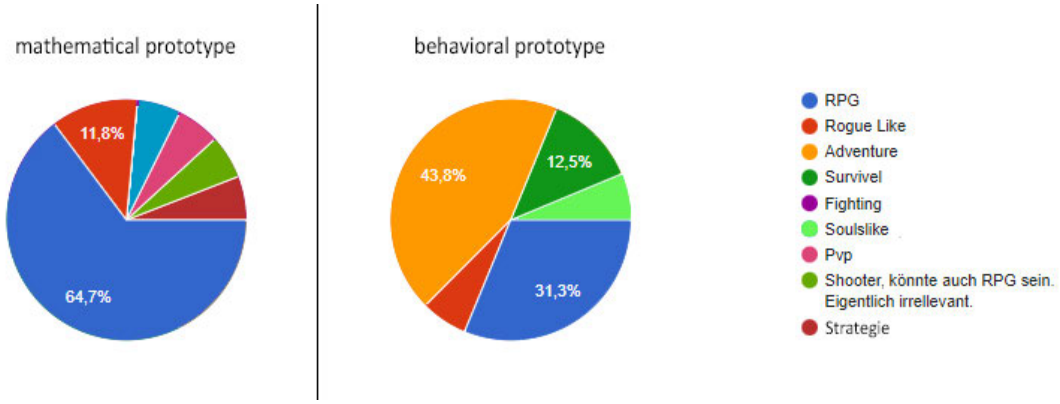
B.3. Evaluation of game behavior with regard to game duration and replay value

Next, participants were asked to imagine the previously played prototype as a finished game in which three different difficulty settings (normal, hard, pro) were modeled on the difficulty levels of the prototype.

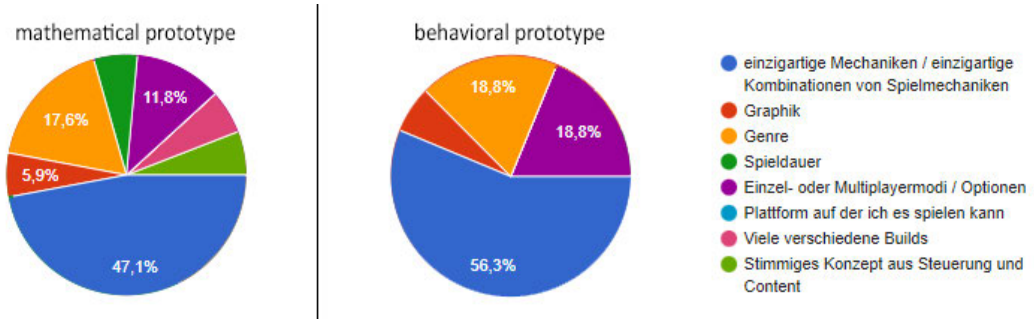
B.3.1. How likely would you be to buy such a game?



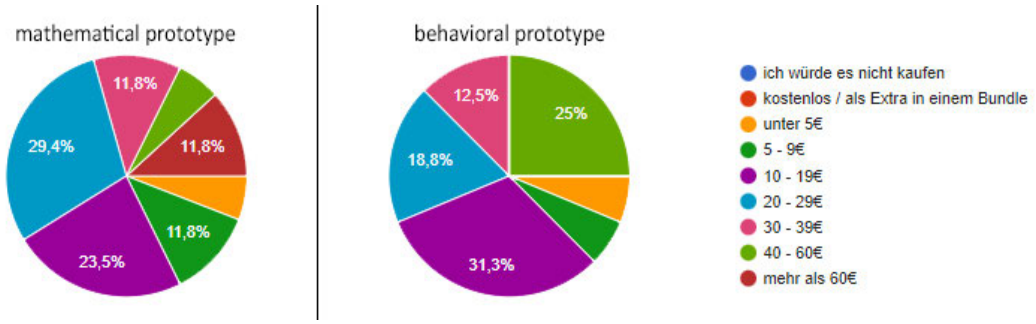
B.3.2. What genre would the game need to possess to make it attractive for you to buy?



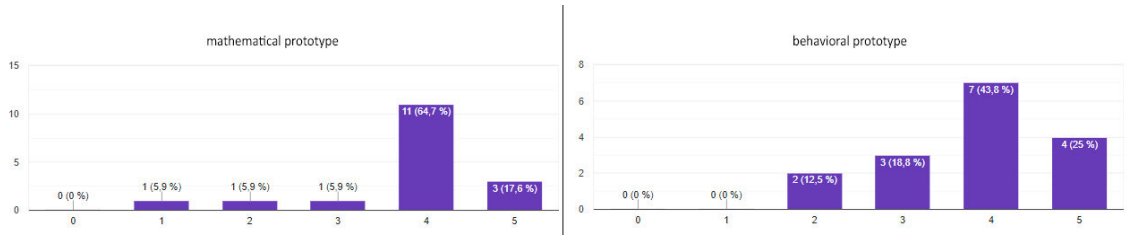
B.3.3. Which criteria would be the most decisive for you when buying such a game?



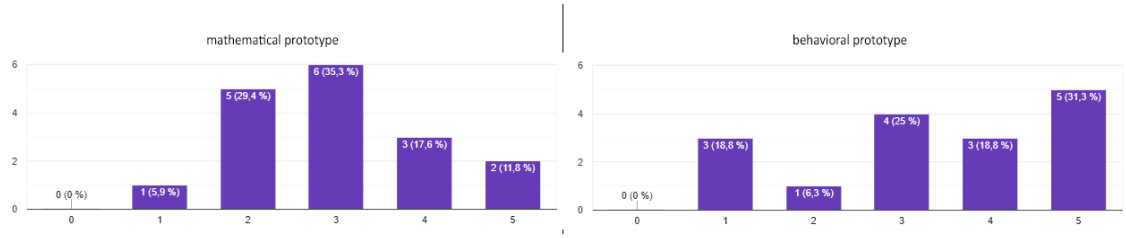
B.3.4. What is the maximum amount of money you would be willing to spend on this game?



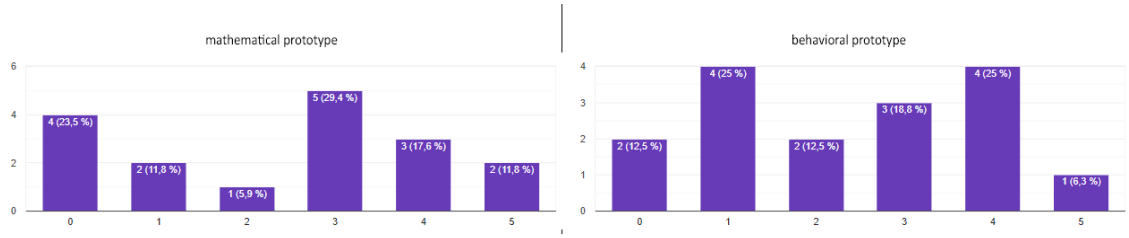
B.3.5. If you already owned the game, how likely would you be to play through it completely?



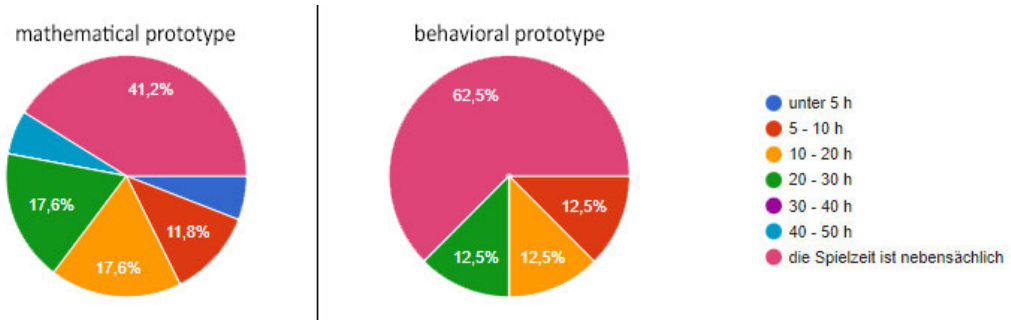
B.3.6. How likely would you be to invest time in the game after finishing the "normal mode"/story for extra content like optional bosses or dungeons?



B.3.7. How likely would you be to replay the game purely on the basis of difficulty progression if the rest of the game (mechanics, graphics, story, etc.) appealed to you?



B.3.8. In regards to replaying the game, what should be the maximum gameplay time required for a "story mode" to get you more excited to play again?



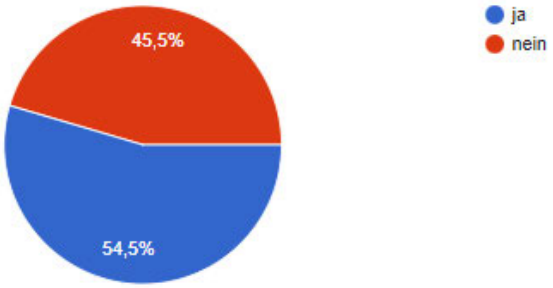
B.4. Meta Artificial Intelligence

Brief description of meta artificial intelligence for the test subjects: In the following, you will be asked questions about a special form of AI that can be used at runtime to adjust the game experience and difficulty to the player's current behavior and abilities. Such an artificial intelligence or instance is called "meta artificial intelligence", because the adjustments are based on collected meta data (e.g. time period over which the player did not get a hit, speed with which the player got through the level, number of deaths before successful completion of the level etc.).

By using meta artificial intelligence, adaptive difficulty levels can be created, matching the required skill level to the player's abilities. Thus, their stress level can be better kept between the state of physical as well as mental overload and underload (also called flow state). These adaptations are of course made as subtly as possible, so that a sense of unfairness is counteracted.

Examples in different application areas to increase the difficulty (of course, the difficulty level can be decreased in the same way vice versa): - In jump and run games, the size of platforms, their spacing, or (for moving platforms) their speed can be adjusted - In combat situations, e.g. the number of enemies to be defeated at the same time, their damage etc. can be adjusted - In survival games, e.g. diseases or environmental influences or even enemies can appear more often or faster, if the preparations of the players guarantee a seemingly effortless survival

B.4.1. Have you ever heard of / played a game that supports adaptive difficulty adjustment?

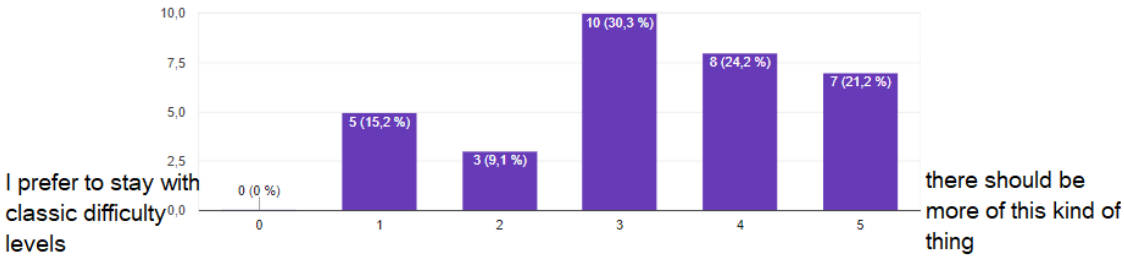


B.4.2. If you answered yes: what game is it?

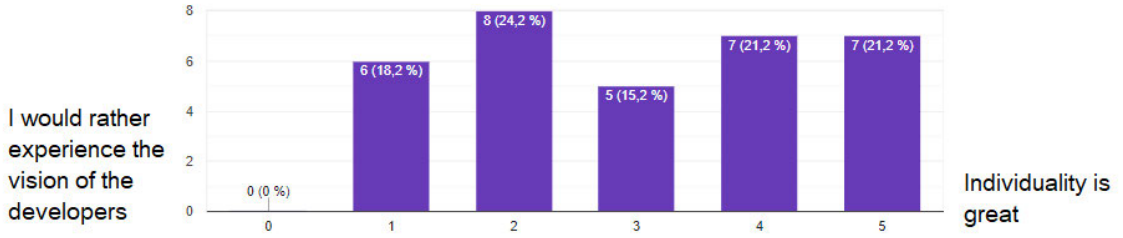
- Alien: Isolation (x2)
- Binding of Isaac
- Bloodborne
- Dark Souls

- Destiny 2
- Dishonored
- Don't Starve
- Fallout 3
- Hades
- Left 4 Dead 2
- Lego Star Wars 2
- Mario Kart
- Metal Gear Solid
- Metal Gear Solid V: The Phantom Pain
- MotoGP
- Need for Speed
- Resident Evil 4
- RimWorld (Storyteller, nur algorithmisch, keine wirkliche KI)
- Rocksmith+ (x2)
- The Forest 2
- viele, moderne AAA-Titel
- Will You Snail?

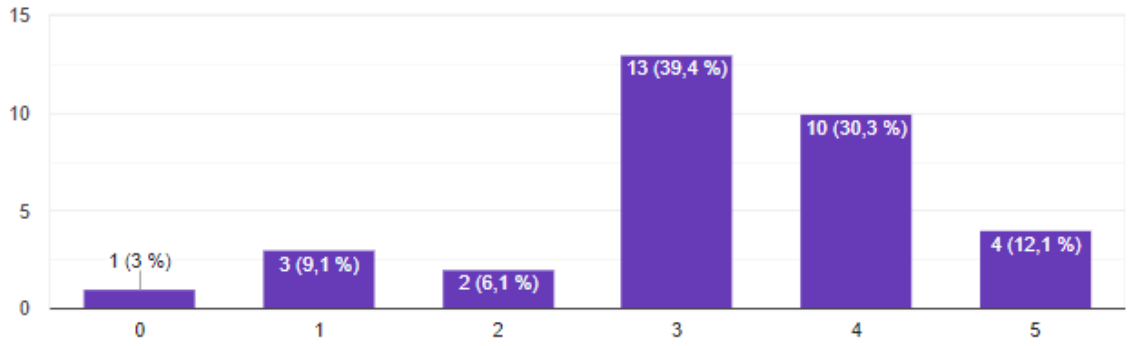
B.4.3. What is your opinion regarding the use of adaptive difficulty in video games?



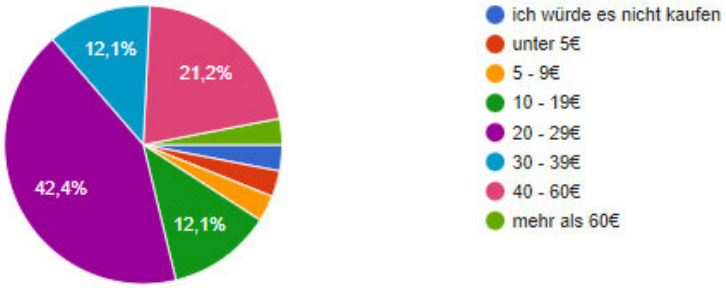
B.4.4. What is your opinion regarding video games that want to offer an individual gaming experience to each player by means of artificial intelligence (e.g. meta AI, procedural content generation, etc.)?



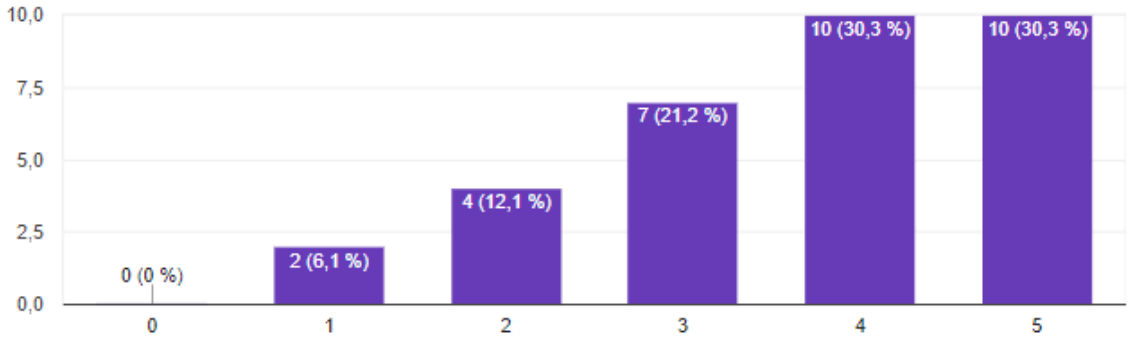
B.4.5. How likely would you be to buy a game that advertises adaptive difficulty?



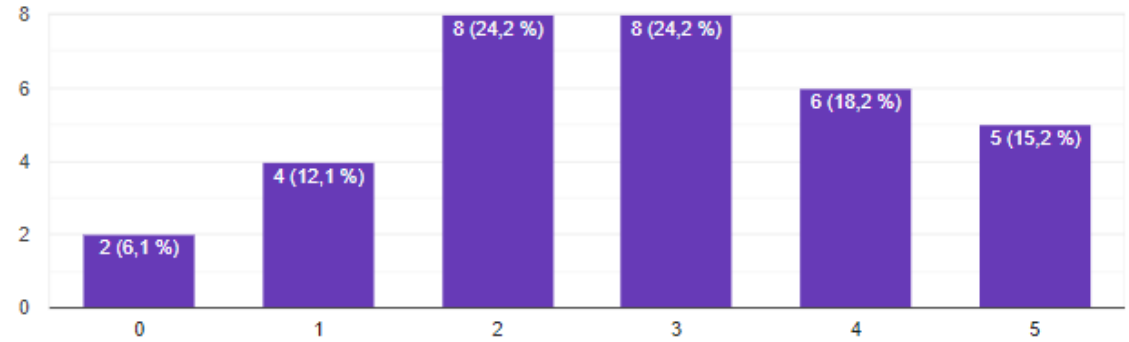
B.4.6. What is the maximum amount of money you would be willing to spend on this game?



B.4.7. How likely would you be to finish a game with an adaptive difficulty level?



B.4.8. How likely would you be to replay a game with an adaptive difficulty level after completing it?



B.4.9. Do you have any more comments on the topic?

- Diese letzte Seite fiel mir relativ schwer, da ich eine große Hürde/Herausforderung bei Meta KI sehe: Die KI muss Dev-seitig irgendwo auch Grenzen aufgezeigt bekommen. Gute Vorbereitung auf Herausforderungen/Kämpfe darf nicht wertlos werden, indem die KI über die Maßen starke Gegenschritte einleitet. Verwende ich im Survival Genre - wie z.B. "The Forest" oder "Green Hell" - viel Zeit darauf mich mit Nahrung, Medikamenten etc. auszustatten, bevor ich in riskantere Zonen aufbreche, nur um dann durch eine (hier wird es wichtig) NICHT limitierte Meta

KI alle paar Minuten "krank" zu werden würde mich dann in Frage stellen lassen, ob es dann nicht klüger wäre genau das Gegenteil von dem zu tun, was die Meta KI erreichen soll: Ein ausgiebigeres Spiel Erlebnis und Wiederspielwert. Mich persönlich würde das Wissen, dass ich mir durch Vorbereitung und Planung nur mehr über die Maßen große Steine in den Weg lege, die ich umgehen kann, dazu verleiten, so wenig wie möglich Content entdecken zu wollen. Bei Kämpfen darf die KI zum Beispiel meiner Meinung nach auch kein Flawless Input Reading auf Basis der Player Actions haben, oder sofort einen Spieler Input + Zusatz abgreifen. So ein Fall könnte besonders bei RPGs ein Immersion Breaker sein, da es dann die Herausforderung wird das Verhalten einer KI durcheinander zu bringen oder zu brechen, statt gegen einen spannenden, vermutlich auch lore-trächtigen Gegner anzutreten und einen Kampf zu erleben, an den man sich gern zurück erinnert. Ebenfalls wichtig sind hier Limits durch Cooldowns oder eine "unsichtbare" Ressource, die auch der KI Grenzen setzt. Eine Meta KI ist eine sehr spannende Sache und ich bin ebenfalls gespannt darauf, wie ihr Einsatz Spiele beeinflussen kann, aber bin ebenfalls davon überzeugt, dass der Datenverarbeitung auch gewisse Grenzen aufgezeigt werden müssen, um nicht eine Art "TASBot" Spiel zu erschaffen, was keinerlei Hebelpunkte mehr für Spieler bietet oder Content Elemente im Spiel vollständig entwertet.

- Adaptive Schwierigkeiten sind meiner Meinung nach nichts gutes. Nehmen wir als Beispiel Final Fantasy 7 Remake Integrade. Ich habe dieses Spiel auf "Normal" begonnen und irgendwann auf "Einfach" umgeschaltet. Weil ich einige Kämpfe als zu schwer empfand und das Spiel hauptsächlich für die Story gespielt habe. Hätte dieses Spiel eine adaptive Schwierigkeit gehabt hätte zwar das Spiel erkannt dass ich mit Bosskampf X nicht gut klar komme und ihn einfacher gemacht mir aber nicht die Wahl gelassen, dass jetzt alle Bosskämpfe so einfach werden. Dieses "rauben" von Optionen sehe ich generell als Problem in modernen Spielen an. Der Spieler soll die Optionen haben sich sein Spielerlebnis so anzupassen wie er möchte. Natürlich nur in Singleplayer spielen. Ich habe kürzlich das Spiel Transport Fever 2 (Spiel indem es um den Aufbau von Logistikketten mit Schiff, Bahn, Flugzeug und LKWs geht) gespielt. Man muss Ziel X mit Geldmenge Y erreichen. Ich habe schnell festgestellt, dass mir dieser "Druck" keinen Spaß macht. Also habe ich mich (mangels Option im Spiel selber) einem Cheatprogramm bedient um mir unendlich Geld zu cheaten. Das ist für andere vielleicht unverständlich aber ich habe das Spiel danach viel mehr genossen. Ich konnte meiner Kreativität freien Lauf lassen

und das Logisitknetz so bauen wie ich es für sinnvoll / ansprechend fand (zum Beispiel lieber Züge statt LKWs bauen) ohne alles optimieren zu müssen etc. Ein gutes Beispiel ist auch Celeste. Dort gibt es einen "Hilfemodus". Der erlaubt es dem Spieler das Spiel selbst angepasst einfacher zu machen wenn nötig. Ohne diesen hätte ich dieses Spiel nie durchgespielt. Ich konnte mir das Spielerlebnis so gestalten wie ich es für richtig hielt.

- Ich finde die Idee der adaptiven Schwierigkeit sehr gut, aber es lässt sich nicht besonders gut auf alle Spiele anwenden. Ein Roguelike z.B. würde für mich mehr Sinn ergeben als ein kompetitives Fighting Game, bei dem es um gleiche Ausgangsbedingungen geht. Kompetitive Spiele allgemein würden aber extrem profitieren, wenn KIs entwickelt werden würden, die ähnlich intelligent spielen wie ein Mensch. Bei Schach gibt es inzwischen KIs, die das Verhalten von Menschen nachahmen könnten. Bei Civilization ist das noch nicht gelungen, sodass die Entwickler den mathematischen Ansatz fahren und den KIs auf höherem Schwierigkeitsgrad enorme Vorteile geben, die sie besonders am Anfang gefährlich machen. Die Entscheidungen der KI sind allerdings weiterhin sehr dumm, sodass, sobald der Mensch den anfänglichen Rückstand auf die KI aufgeholt hat, er das Spiel immer gewinnt. Dadurch sind Spiele gegen die KI ab einem bestimmten Skill eher uninteressant, weil es eigentlich nur darum geht, das early-game irgendwie zu überstehen. Eine KI, die hingegen entweder eher so "denkt" und und handelt wie ein menschlicher Spieler würde sehr reizvoll sein. Man könnte aber auch überlegen, ob eine adaptive mathematische KI nicht auch funktionieren könnte, die die Vorteile der KI im Late-Game immer weiter steigen lässt, sodass ihre "Dummheit" besser kompensiert wird. Ein super spannendes BA-Thema, darüber könnte man sich auf jeden Fall ewig unterhalten.

- Ich finde die Idee eines adaptiven Schwierigkeitsgrades in nahezu allen Genres super, außer Survival. Dort geht es darum, sich möglichst gut auf Unbill vorzubereiten, und zu wissen, dass man, ist man schlecht vorbereitet, vom Spiel "verschont" wird, macht für meine Begriffe den Sinn des Spiels zunichte. Hingegen kann ich mir vorstellen, dass eine clever verwendete adaptive KI dem Spiel zusätzliche Würze verleihen könnte, indem Veteranen häufiger oder in größerer Stärke gefährlichen Situationen ausgesetzt sind, und sich somit gut fühlen können mit der Menge an getroffenen Vorbereitungen. Ein Survival Spiel sollte den Spieler aber niemals "schonen".

- Adaptive Schwierigkeits grade sind sehr viel schwerer "zuzuschneiden" für eine gewollte erfahrung. Wie im oben genannten Survivalgame Beispiel; ich würde es unfair finden wenn ich plötzlich mit Gegners überrannt werde/ich genauso viele Schwierigkeiten habe zu überleben trotz der Ressourcen die ich gesammelt habe. Außerdem finde ich die Gegenüberstellung von "Individualität" und "Vision der Entwickler" oben als unpassend. Die Vision der Entwickler kann eine individuelle Erfahrung sein, genauso wie zu viele Ressourcen in einem 'festen' Schwierigkeitgrad zusammeln die Vision der Entwickler zerbricht

- Die Antworten sind natürlich auch sehr abhängig davon, was das Spiel sonst noch zu bieten hat, mir wären wahrscheinlich viele anderen Faktoren Wichtiger, als das Kampfsystem. Davon hängt aber auch ab, ob ich ein Spiel durchspielen und wie viel Geld ich dafür ausgeben würde. Ich finde es einerseits natürlich cool jedem individuell ein Spielerlebnis zu bieten, da ja auch jeder unterschiedliche Fähigkeiten hat und das Spiel würde sich ja auch mitentwickeln. Jedoch finde ich es auch gut, wenn klassisch jeder die gleichen Grundvorraussetzungen hat.

- Während des Spielens der 3. Stufe des Prototypes fing ich nach einer Weile an, bewusst den Enemy-Lock abzuschalten und schräg seitlich neben den Gegner zu zielen. Durch den Radius des Schwerthiebs treffen Angriffe dieser Art trotzdem, während die Wahrscheinlichkeit das der Gegner einen trifft deutlich geringer ist, wenn man genug Distanz wahrt. Das fühlte sich zwar etwas cheesy an und reduzierte den Spielspaß, machte es mir aber deutlich einfacher Stufe 3 abzuschließen.

- Ehrlich gesagt habe ich mich damit bisher fast gar nicht befasst, also sei es mir hoffentlich verziehen, wenn ich aus dem Bauch heraus antworte. Zudem würde ich die Entscheidung dafür, ob ich das Spiel kaufen würde, nicht von den Schwierigkeitsgraden abhängig machen, sondern von Gameplay, Ingame-Setting & Look. Außerdem habe ich mich für die "Unter-5€"-Kaufoption entschieden, weil ich Spiele meist im Sale kaufe oder bei Aktionen gratis bekomme.

- Level 3 habe ich mehr oder minder durch einen Exploit gewonnen. Ich habe nicht das Lock-On Feature verwendet, sondern habe den Gegner mit der Kamera angeschaut, dann bin ich so vor ihm weggelaufen, dass er leicht rechts hinter mir steht. Danach habe ich einen Angriff gestartet, der den Gegner beim Ausholen trifft und Schaden macht. Meist konnte mich der Gegner dabei nicht mit eigenen Angriffen treffen.

- Ich hatte immer das Gefühl, dass man bei adaptiven Schwierigkeitsgraden als Spieler etwas schummeln kann. So kann man z.B. sehr schnell spielen und bestimmte Aktionen auslassen, sodass die KI mit der Anpassung nicht richtig hinterherkommt oder man kann mit Absicht in bestimmten Situationen schlecht spielen, um die Schwierigkeit zu verringern.
- KI Gegner welche Verhaltensmuster im Bezug auf den Spele (und seine Spielstärke) verändern ist erstrebenswert, in Sachen Welt und Level design (in Form von Rätseln, Shortcuts, Gegnerplatzierung, Architektur und sonstigen Ereignissen gern Klassisch "starr" halten
- Die adaptive Schwierigkeit ist für mich weder ein Kaufargument, noch ein Grund oder Hindernis ein Spiel zu spielen. Meiner Meinung nach ist es egal, ob das Spiel feste oder variable Schwierigkeit bietet, wenn es ein "gutes Spiel" ist.
- Es kommt bei den letzten beiden Fragen sehr darauf, ob mich das Spiel von der Graphik und der Story anspricht, falls dies zu 100% der Fall sein sollte, würde ich es definitiv nochmal spielen und auch definitiv beenden.
- adaptive Schwierigkeit ist genauso wie Schwierigkeitsgrad und Barrierefreiheit wichtig und geht mit beiden Hand in Hand, muss aber transparent sein, d.h. evtl. auch an- und ausschaltbar sein insofern praktizierbar
- die Vorstellung eines sich an den spieler anpassenden spieles ist sehr verlockend aber trotzdem sollte es die möglichkeit geben, verschiedene vorgefertigte schwierigkeitsgrade des entwicklers spielen zu können
- Ich finde es super interessant und forsche auch selber schon daran. Mich interessiert beispielsweise auch noch eine meta ai auf höheren meta-Ebenen als nur dem direkten Geschehen im Spiel selbst.
- Alles sehr stark abhängig von der Art des Spiels, passt mMn besser bei einem Roguelike oder Survival Game als bei einem RPG.
- Super interresantes Konzept, stelle ich mir grade bei Mechaniken von Spielen wie Dark Souls oder so mega gut vor
- Sehr nice idee

C. Collected Data during Playtest

While play testing one of the two prototypes (behavioral and mathematical), play data was collected. This includes which scene or stage was played in which order, the successor of the respective fight as well as the time it took for one of the combatants to die. The data that was collected during these tests are documented in table C.1 for the mathematical prototype and in table C.2 for the behavioral prototype.

Table C.1.: Playdata collected while the mathematical prototype was tested

avg. time Stage 1	times played	when cleared	avg. time Stage 2	times played	when cleared	avg. time Stage 3	times played	when cleared	total playtime
24,92	1	1	32,93	11	10	9,38	1		11:43
28,98	1	1	31,37	3	3	17,17	8		10:04
28,53	1	1	41,23	2	2	12,58	57	57	14:37
51,71	1	1	60,92	1		0			1:56
47,87	1	1	47,82	1	1	27,48	25		13:28
15,55	1	1	24,83	1	1	25,9	6	6	03:22
18,94	1	1	38,25	2	2	16,89	35	29	12:25
15,78	1	1	44,96	4	4	29,78	15	15	11:17
29,73	3	1	39,32	2	2	20,88	25	25	12:19
54,25	1	1	69,15	1	1	42,26	7	4	7:51
24,18	1	1	53,01	1	1	16,48	1		1:38
55,89	2	1	44,73	12	5	31,53	18	18	21:37
28,16	1	1	45,01	3	3	16,27	32	32	12:04
28,84	9	1	40,33	32	2	18,99	29		23,21
42	1	1	49,34	2	2	16,8	4		3:58

Table C.2.: Playdata collected while the behavioral prototype was tested

avg. time Stage 1	times played	when cleared	avg. time Stage 2	times played	when cleared	avg. time Stage 3	times played	when cleared	total playtime
20,03	2	1	32,9	2	1	65,65	6	1	9,30
32,08	1	1	47,19	1	1	65,09	1	1	2:34
62,24	3	1	57,15	2	1	56,8	3	1	9:34
56,57	1	1	49,46	1	1	40,48	2	1	3:48
34,24	3	1	56,41	1	1	45,82	18	1	17:18
27,33	2	1	31,21	2	1	38,07	3	1	4:48
16,95	1	1	22,92	1	1	21,35	1	1	1:06
23,1	1	1	32,46	1	1	44,62	1	1	1:47
79,25	1	1	28,51	1	1	57,64	1	1	2:51
25,44	2	1	45,1	3	1	25,63	3	1	5:15
15,76	1	1	20,48	1	1	39,34	1	1	1:21
25,33	1	1	30,79	1	1	21,05	1	1	1:36
36,64	10	2	24,58	3	1	38	8	1	17,59
22,15	2	1	32,08	1	1	73,34	2	2	3:54
20,95	1	1	50,17	1	1	35,88	1	1	1:53
29,61	1	1	38,09	1	1	31,41	2	1	9:26

Declaration of Authorship

I hereby declare that the thesis submitted is my unaided work. All direct or indirect sources are acknowledged as references. This paper was not previously presented to another examination board and has not been published.

Mittweida, the 14th August 2022

Tina Kießling