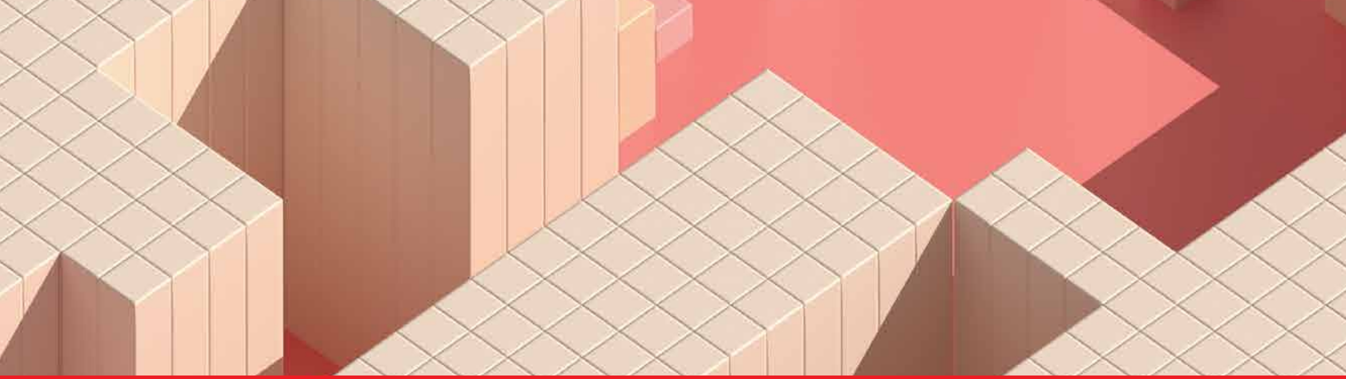
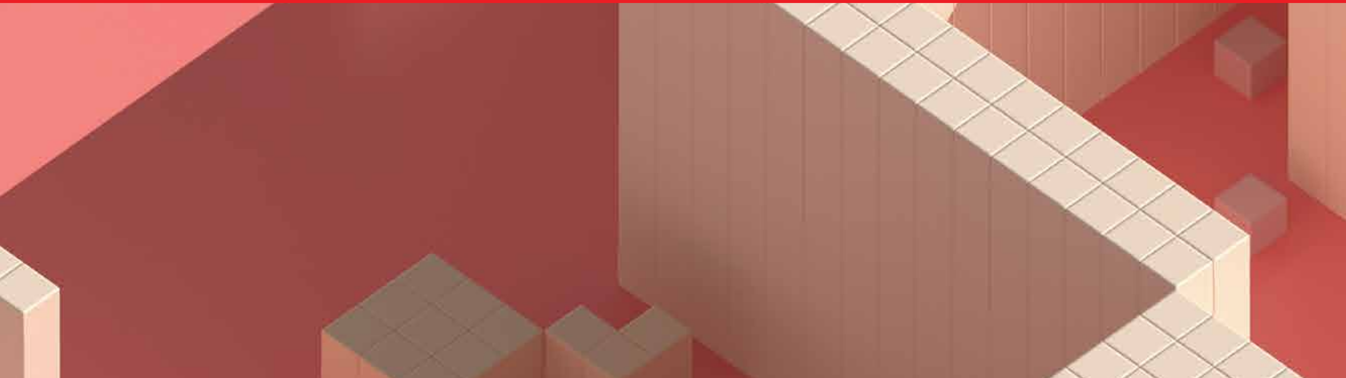


IntechOpen

Computer Science for Game Development and Game Development for Computer Science

Edited by Branislav Sobota and Emília Pietriková



Computer Science for Game Development and Game Development for Computer Science

Edited by Branislav Sobota and Emília Pietriková

Published in London, United Kingdom

Computer Science for Game Development and Game Development for Computer Science
http://dx.doi.org/10.5772/intechopen.1000364
Edited by Branislav Sobota and Emília Pietriková

Contributors
Branislav Sobota, Emília Pietriková, Demos Parapanos, Mifrah Ahmad, Hsuan-min Wang, Cheng-Lun Tsai, Chuen-Tsai Sun, Branislav Sobota, Paula Escudeiro, Márcia Campos Gouveia, Kinga Kovácsné Pusztai, Xavier Fonseca

First published in London, United Kingdom, 2023 by IntechOpen
IntechOpen is the global imprint of INTECHOPEN LIMITED, registered in England and Wales, registration number: 11086078, 5 Princes Gate Court, London, SW7 2QJ, United Kingdom

British Library Cataloguing-in-Publication Data
A catalogue record for this book is available from the British Library

Additional hard and PDF copies can be obtained from orders@intechopen.com

Computer Science for Game Development and Game Development for Computer Science
Edited by Branislav Sobota and Emília Pietriková
p. cm.
Print ISBN 978-1-83769-733-5
Online ISBN 978-1-83769-732-8
eBook (PDF) ISBN 978-1-83769-734-2

Meet the editors

Branislav Sobota obtained an MSc with honours from the Department of Computers and Informatics, Faculty of Electrical Engineering and Informatics, Technical University of Košice (TUKE), Slovakia, in 1990. He defended his Ph.D. in 1999 and his habilitation thesis in virtual reality and computer graphics in 2008. He is an associate professor in the Department of Computers and Informatics, TUKE. His scientific research focuses on computer graphics, parallel computing, and virtual reality and related technologies.

Emília Pietriková is an associate professor in the Department of Computers and Informatics, Technical University of Košice (TUKE), Slovakia. She received her MSc and Ph.D. in Informatics from TUKE in 2010 and 2014, respectively. The subject of her research is source code analysis, programming techniques, computer game design and development, and quality of higher education.

Contents

Preface

In the fascinating landscape of evolving computing technology, few facets of our digital world captivate us as profoundly as computer games. In this era of digital addiction, it is not enough to merely play these games; we must delve into their creation. The development of computer games is a captivating journey that melds seamlessly with the world of computer science. It is a journey that encompasses programming, software engineering, computer graphics, databases, artificial intelligence, and the intricate physics that underpin these virtual worlds. As computer games continue to evolve, growing in complexity and sophistication, the role of computer science in their development becomes increasingly pronounced. This book is a voyage into this dynamic realm, where technology meets creativity, and where the next generation of digital innovations is forged. Throughout this book, we will explore the intricate relationship between computer science and game development. We will uncover the connections between these two domains, shedding light on how technology and creativity converge to shape the world of computer games. Together, we will navigate this dynamic intersection where innovation and imagination merge.

We would like to express our sincere gratitude to all the authors and co-authors for their contributions. The successful completion of this book has been the result of the cooperation among many people. We would especially like to thank Publishing Process Manager Ms. Nina Miocevic at IntechOpen for her support during the publishing process.

Branislav Sobota and Emília Pietriková
Technical University of Košice,
Košice, Slovak Republic

Chapter 1

Introductory Chapter: Game Development and Computer Science

Branislav Sobota and Emília Pietriková

1. Introduction

Nothing affects a person's life more than the games he played in his childhood. Games open a wonderful world of colors, light, secrets, materials, and relationships. The feeling of the game carries with us even into adulthood, and our desire to play remains in the subconscious and regularly emerges. In the colossal change of the world caused by computers, it remains unchanged, but its modification appears—computer games [1]. In recent years, they have incredibly influenced children and adults.

The development of computer technology and its impact on our lives are striking. Entertainment, especially computer games, is no exception in this sense. Playing computer (including electronic) games is a substantial percentage of this area. Of course, the development of these games is closely related to this. The development of computer games requires in-depth knowledge in the field of computer science, starting with programming, software engineering, or computer graphics (in some cases, in this sense, also virtual reality), through databases, artificial intelligence, and ending with, for example, the physics of games [2]. In short, as computer games become more complex, sophisticated, and immersive, the role of computer science in game development becomes increasingly important.

In the secondary view, computer science is integral to every phase of computer game development. First, it provides computer game developers with the necessary tools to create, optimize and debug computer games, develop their game mechanics and gameplay, design interactive environments and characters, and create visual and sound effects [3]. Secondly, there is a close connection with the business aspect of computer games, including game marketing, distribution, or advertising [4].

Following that, from the view of current game systems, it is necessary to address several areas (**Figure 1**). This applies mainly to the following areas (in no order of importance): graphics, sound, game logic and design, user interface (depending on the type of game also 3D interface, including player motion tracking or, e.g., his gesture sensing), sometimes different kinds of simulations or training modes, including the possible use of digital twins technology (if there is a game session with a potential real world), possible in-game physics, dynamics processing including scripting, the most immersive gaming experience possible, single-user or multi-user game mode including a possible cooperative mode, educational or research potential of the game or possibly data processing, artificial intelligence, and of course the economic aspect.

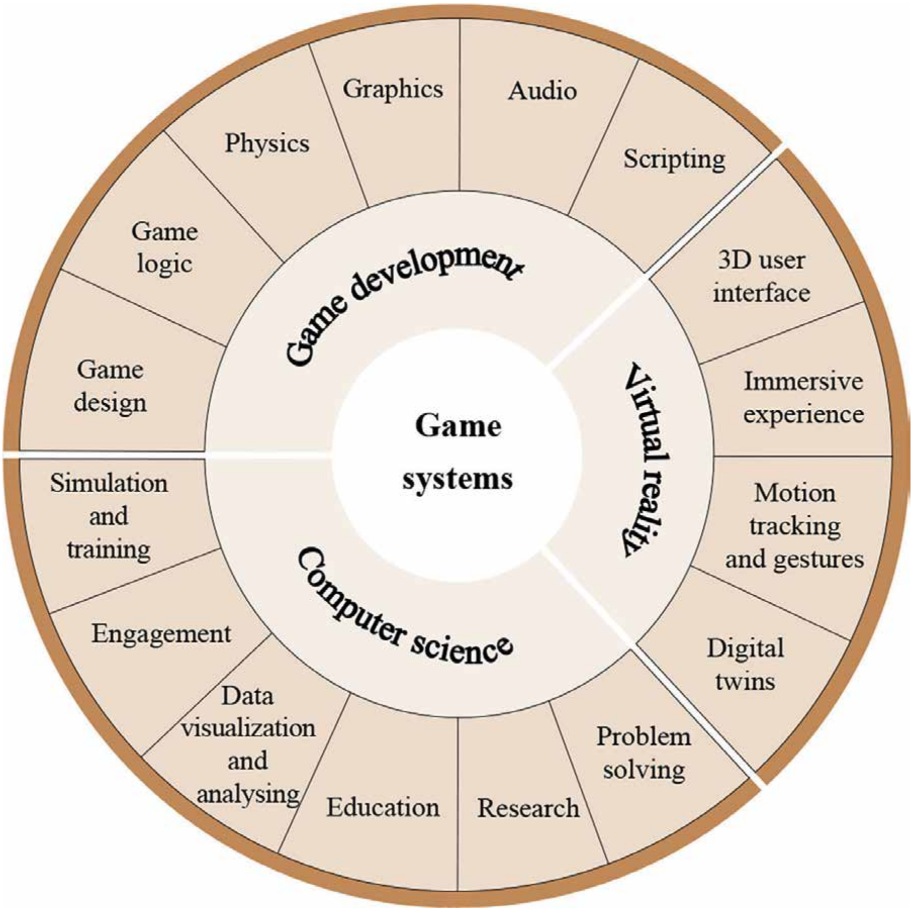


Figure 1.
Game development, virtual reality, and computer science relationship.

In the opposite direction, the development of computer games also significantly influenced computer science as a discipline, mainly by pushing the boundaries of the current state of technology and supporting innovations in various areas. Computer games played a crucial role in introducing new technologies at the hardware level, such as graphics processing units (GPU and GPGPU technology), multi-core processors, or virtual reality, which further accelerated the progress of computer science. At the software level, computer games and their development have an impact on the fields of computer science mainly in the following ways:

- *Computer graphics*: Computer graphics and animations are primarily affected by computer games. They play an important role in game development and require programmers to use algorithms for transformations, rendering, shading or lighting effects, etc. This led to progress in this direction, but also in computer vision and image processing.

- *User interface and user experience (UI/UX)*: An engaging and intuitive user interface/UX in games is essential for comfortable communication with a computer and thus constitutes an integral component of the success of a computer game. This has led to advances in human–computer interaction (HCI), interface design, and the use of user experiences in computer science. The impact of virtual reality and associated technologies in this direction is significant.

- *Computer networks and multiplayer*: Some computer games often involve communication and interaction between players, which requires network connectivity, communication protocols, security, and reliability. This led to progress in computer networks, distributed systems, and cyber security.

- *Formalization and optimization techniques*: Game development usually requires formalization and optimization techniques, such as processing game rules, game logic or reducing the number of calculations needed to render graphics, improving loading times, and improving the gaming experience. Formalization, especially with game theory, represents a direct link and enables more convenient algorithmization.

- *Artificial intelligence and genetic algorithms*: This part of computer science cannot be neglected nowadays. Some aspects of artificial intelligence have long been implemented, such as controlling characters in games, making decisions or moving them, and finding possible movement trajectories. Genetic algorithms are helpful, for example, for describing the behavior of some entities in the game.

The primary relationships between the development of computer games and computer science from the point of view of computer games are shown in **Figure 1**. Many aspects have been mentioned in the previous one, and we will cover some of them in the following text. Due to the innovative procedures brought by virtual reality and associated technologies, VR is also included in this picture.

Looking to the future for certain types of games, VR or related technologies are often implemented because they have the potential to push the development of computer games to an entirely new level [5]. In essence, the VR system represents an interactive computer system, creating the illusion of a non-existent, only synthesized space at the given time, or even more precisely, we can talk about the so-called perfect simulation in an environment of close human–computer system relation. In this context, the development of computer games on this basis includes creating digital games for entertainment, educational, or training purposes [6] that rely heavily on graphics, visuals, and interactivity. Thanks to advances in virtual reality technologies, game developers can now create games that offer the user an immersive and interactive experience. Players can now become active in the game's story and explore living worlds that feel incredibly realistic. Thus, virtual reality enables complete immersion and blurs the line between the player and the game. The concept of Metaverse (https://about.meta.com/metaverse/) can bring the development of computer games to an even higher level. Metaverse refers to a virtual shared/collaborative space where users/players can interact with a computer-generated environment and other users/players in real time.

2. Conclusion

The ancient Roman demand for "bread and circuses" survived for centuries. Human playfulness got wings with the advent of computer games. They have become so popular among the population that masses of avid gamers (mainly children) succumbed to them quickly. The unceasing demand for computer games and the fact that more and more people are involved in this segment, creating, improving, researching, and analyzing computer games from a scientific point of view are behind the emergence of scientific directions oriented to computer games. Their mutual interaction with computer science reaches the stage where part of the results of this interaction is usable and is used in practice in areas such as medicine, therapy [7], services, transportation [8], or education [9]. Its further potential is enormous, which is good news for players, developers, and scientists [10]. Ultimately, we have an exciting future ahead of us in this area.

Acknowledgements

This work has been supported by Slovak KEGA Agency under grant no. 048TUKE-4/2022: "Collaborative virtual reality technologies in the educational process," under grant no. 002TUKE- 4/2021: "Implementation of Modern Methods and Education Forms in the Area of Cybersecurity towards Requirements of Labour Market" and the APVV grant no. APVV-21-0105 "Trustworthy human–robot and therapist–patient interaction in virtual reality."

Author details

Branislav Sobota* and Emília Pietriková
Technical University of Košice, Slovakia

*Address all correspondence to: branislav.sobota@tuke.sk

References

[1] Predescu A, Mocanu M. A data driven survey of video games. ECAI Conference. Danvers, USA IEEE; 2020. pp. 1-6

[2] Mansouri B, Roozkhosh A, Farbeh H. A survey on implementations of adaptive AI in serious games for enhancing player engagement. ISGS Conference. Danvers, USA: IEEE; 2021. pp. 48-53

[3] Pichlmair M, Johansen M. Designing game feel: a survey. IEEE Transactions on Games. 2022;**14**(4):138-152

[4] Rebualos R, A et al. Customer Preferences for Online Games: A Conjoint Analysis Approach with Market Segmentation. 6th ICSeB 2022 Conference. New York, USA: ACM; 2022. pp. 33-39

[5] Kao D, Magana AJ, Mousas C. Evaluating tutorial-based instructions for controllers in virtual reality games. ACM on Human-Computer Interaction. New York, USA: CHI PLAY, ACM; 2021. art. 234

[6] Dorrigiv M. Incorporation of serious games into higher education: A survey. ISGS Conference. Danvers, USA: IEEE. 2021. pp. 86-90

[7] Chan L et al. TOMO VOX: Exploring Cozy games and character-driven Storyworlds for Pediatric voice therapy. New York, USA: CHI PLAY, ACM; 2022. pp. 127-132

[8] Hariz M, B et al. Game theoretic approach for a multi-mode transportation in smart cities. Danvers, USA: ISNCC Conference; 2020. pp. 1-6

[9] González-Campos J, S et al. GTCards: A video game for learning geometric transformations: A cards-based video game for learning geometric transformations in higher education. TEEM Conference. New York, USA: ACM; 2021. pp. 205-209

[10] Grace LD. Social impact games, a probable future illuminated by looking back. Games. Research and Practice. 2023;**1**(1):1-3

Chapter 2

Make It Fun for Everyone!

Demos Parapanos

Abstract

This chapter will provide an overview and strength of games, as well as examining the element of fun influencing gamers' behavior in games. The development of sophisticated algorithms and the increased coverage of the internet help electronic games succeed and increase, overflowing the traditional boundaries in which they were traditionally confined. As the game industry enjoys continual and rapid growth in the market, it is important to examine the features that make games so attractive and keep players coming back for more. Professional game designers often take it for granted that people just want to play their games, rarely examining the psychology of their gaming audience. Hence, the aim of this chapter is to focus on understanding the meaning of fun in games and the interaction with algorithms, which will create diagrams to predict gamers behaviors in a game. Personal data collected through algorithms can be a tool for game designers to develop future games for specific audiences. The rationale is that examining the psychology behind the subjective element of fun will help increase the interaction between gamers and the game for the algorithms to create the appropriate environment for the player.

Keywords: games, gamers, fun, gamers' profile', games' diagram, engagement in games

1. Introduction

Throughout the years, video games have become a multi-billion-dollar media industry, reporting profits of more than the movie and music industries combined [1]. Sajid et al. [2] understood that video games and gaming industry reported revenue of $12 billion per year, with Hilgard et al. [3] adding that particularly in the US the video game industry is among the fastest-growing sectors of the economy. In 2014 Cox [4], compared the video game industry revenue with other entertainment industries such as movie and book industries to investigate the revenue difference. The blockbuster movie at that time ("Harry Potter and the Deathly Hallows") recorded earning $169 m in revenue during its opening weekend, while the book "Harry Potter and the Deathly Hallows" generated $220 m in the first 24 hours of release. To highlight the difference between them, two industries and the gaming industry Cox [4] presents that in the same year, a game called "Call of Duty: Modern Warfare 3" raised $400 m in revenue in just the first day of release; when the same video game generated $1bn in revenue within the first 16 days, overcoming the previous entertainment record set in 2009 by the film "Avatar" within the first 17 days of release.

Gaming has become a very popular activity for both adults and adolescents and a major leisure activity in contemporary societies [5]. Even though most video players are aged between 18 to 34 years old (40%), 27% of players are being overage 50, leading to the average player being 35 years old [6] indicating that there is also a big percentage of players over the age of 35 spending time on video games, and highlighting that players cannot be grouped into one age demographic. Reports from the Interactive Software Federation in Europe show that more than half of Europe's population plays video games and nearly half of them are female (46%) [7], showing that there are also no demographic differences between gender. Studies in the US show that 80% of people aged 18–34 play video games for more than 3 hours a week [6] when the average time spent gaming per week is estimated to be around 8 hours and 42 minutes [7].

As video games have grown in universal popularity, they have also grown in diversity [3], empowered by the development of technologies, such as smart mobile devices, advanced algorithms, and internet accessibility [8]. Mobile games use generational innovations to become available on multiple platforms [9]. The advances in the current digital era and the increased coverage of the internet help electronic games succeed and increase [10], overflowing the traditional boundaries in which they were confined [11]. As the game industry enjoys continual and rapid growth in the market, it is important to examine the features that games are so attractive and keep players coming back for more [12]. Professional game designers often take it for granted that people just want to play their games, rarely examining the psychology of their gaming audience [13]. Previous studies (see [6, 13, 14]), investigated behaviors of players when playing games creating a typology of them. However, previous literature with limited exceptions (see [15]) has not investigated in depth the meaning of fun for the players ecosystem. Hence, the aim of this chapter is to focus on understanding the meaning of fun in games and the interaction with algorithms which will create diagrams to predict gamers behaviors in a game. To achieve the aim this chapter will review relevant literature and previous research that focused on behaviors of players in games and their preferences in several types of games. Using some of these findings, this chapter creates a diagram to measure players' behavior in a game.

2. Games

2.1 Power of games

McDaniel and Forsyth [16] identify that video games are popular activities among both youth and adults. Just in the USA about 97% of teenagers appear to play contributing toward video games and gaming industry to report revenue of $12 billion per year [2]. Similarly, reports since 2015 announced high percentage in countries such as Australia, where 95% of adolescents have access to at least one video game device in their home through tablet, smartphone, or personal computer (PC) [17]. Even though the attachment of teenagers with video games has been a stereotype over the years, studies like Adachi and Willoughby [18], report that video gamers age demographic is expanded to older adolescents and emerging adults. In their report they understand that 81% of 18–29-year-old Americans play video games and half of them play games at least a few times a week [18]. Rogers [19], supports the previous arguments reporting that over 183 million people, (or 49%) of adults in America, play video games, supporting that video games attract and engage individuals at any stage of their life.

Becoming more varied, socially inclusive, and accessible, video game sales now surpass the US$43 billion mark and boast larger revenue streams than digital music and blockbuster films combined [20]. Back in 2016, the video game market in the USA was valued at over $17 billion mainly through three companies, namely Microsoft, Sony and Nintendo, monopolizing the home video game industry [21]. In Germany, the gaming industry is ahead of other cultural and media industries with more than 6.2 billion euros in revenue a number to grow 19% in 2023 [22]. The popularity of video games in the past decade is also empowered by the rapid development of smart mobile devices [8]. Apple's "App Store" and Google's "Google Play" announced that 19.2 billion mobile games were downloaded in 2016 and users spent on average $40 per device on premium applications, dominating consumer spending [23]. Two years later Apple Store and Google Play both announced increases in numbers to 21.8 billion with forecasts for continued growth in the future [24].

The revenue numbers are impressive compared with other entertainment industries such as cinema and books, but still more impressive is the engagement gaming industry provides. The 'average' video game player has been engaged with the activity over several years, resulting in more people from different age groups engaging with playing games. In their research Smith et al. [17] found that Australian adolescents play computer games anywhere between 2 and 18 hours per week. Similarly, American teens play video games around 13 hours per week [16]. Increased time spent in video games also affects adults looking at a report from Yu and Chan [25], showcasing that older adults consider themselves gamers playing video games for pleasure or even training activities. Gamers play some kind of video game for over 15 hours a week, where non-gamers play video games less than 4 hours [26]. Overall, statistics show that since 2011, people were already spending an average of three billion hours a week gaming with the number increasing exponentially through the years [10]. The abovementioned report supports the argument that gamers are engaged with games over a period.

2.2 Games and algorithms

As video games exhibit progressively expansive game environments, there has been growing interest in employing generative computational algorithms to mitigate the cost of authoring game content [27]. The advantage of computational techniques lies in the promise to reduce the involvement of a human designer, hence enabling financial advantages due to smaller development teams, but also content that would otherwise be impossible. Advanced algorithms generate content more flexible than content generated by hand, providing opportunities for the game to adapt to the unique preferences of individual players, making that content more personal. For example, games generated by advanced algorithms evolve software that can recognize behaviors and patterns and adapt to players preferences. In open map games, the fitness of a given weapon design is inferred from the behavior of the player during the game progress; if the player uses a weapon frequently, similar weapons are made available, and where a weapon is left unused, it appears less frequently [27].

The outcome of this evolved technology means that the content is highly personalized, promoting individual preferences and creating engagement and sustainability, hence user is more likely to be engaged with the system as they recognize their preferences and problem-solving mechanics in the game. The success of the previous example derives from the fact that the user is continually using the system giving plethora of information to the platform, and this can only be achieved if the user is engaged with the system providing data to the system to be utilized in future activities and tasks. Despite the evolution of technology and advanced algorithms, the success of the system relies on engagement with the player, so data to be absorbed and patterns to emerge.

The relationship between computational techniques and the engagement with the game seems to be interdependent since players engagement is needed for the game to collect enough data to acknowledge a profile for the player, and on the other hand, these data will be transformed into mechanics and esthetics that will make the game personalized and engaging. Considering that gaming is optional activity [16], the engagement of the gamer with the game is not given, hence for gamers to return in the game, the system will have to create previous enjoyable experience. In the year 2010, Hoffman and Nadelson [28] found that engagement in games is translated to achievement, motivation, and task persistence. Two years later, Vorderer and Jennings [13] further add that the main tool enhancing the engagement in games and differentiating them with other entertainment industries (such as movies) is the element of control that games provide to the user creating a sense of immersion and interactivity.

Looking at the two big entertainment industries, it is obvious that the main difference between them industries is the element of control that games provide to the user due to the interaction between the gamer and the game through advanced algorithms that take advantage of that. Unlike movie viewers, game players not only process information provided by the medium but they also contribute to the quality and progress of the media product itself, with their decisions and actions determining how a game looks, develops, and ends. This is a result of a continuum interaction between the game and the gamer, and the data collected informs algorithms to personalize the experience and make it more appealing and fun. Despite the acknowledgment of the interdependence between the algorithms and the engagement, industry understands that [13] professional game designers often take it for granted that people just want to play their games, rarely examining the psychology of their gaming audience. However, the rising global popularity of video games is occasionally associated with research interest in solving issues related to video games such as decision-making, movement, strategy, and pathfinding through algorithms [29]. However, it is not only the technological advances of games that enhance the engagement between the gamer and the game [30] because from the psychological standpoint, the concept fun is seen as the most important and necessary factor in game playing [16].

2.3 Engagement in games

Professional game designers often take it for granted that people just want to play their games, rarely examining the psychology of their gaming audience [13], focusing on developing the technology and esthetics in the game rather than the patterns that create engagement. In 2015 Siemens et al. [31] highlight that importance for individuals acknowledging that even though it is within human nature to like games, not everyone likes the same kind or style of games. Marczewski [32] agrees on the subjectivity of the topic stating that despite the capability of designing games, serious games, or gamified systems without knowing who the target players are, it is more likely to achieve an engaging experience when the target players are identified first. Melodia et al. [33] then argue that the captivation of video games appears to lie in their potential to fulfill those psychological needs also refer to as motives to play. Lastly, Parapanos and Michopoulou [34] also agreed on the importance of recognizing the psychological need of players' motives when playing games for better engagement.

According to Hilgard et al. [3], the subjective norm behind players motives to play games inspired researchers, players, and game developers measuring individual differences in game motivations and preferences. Human motivation refers to the internal process that activates and maintains physical and psychological activity and impacts the direction and strength to move toward goals [6], highlighting the subjectivity of the topic. As early as 2004 Bartle [15], first recognizes that in a non-entertainment industry, the answer behind motives toward an activity is hidden behind the clear and easy answer of the mandatory nature of that action. Since 2008 gaming industry understood that in video games, players are not required to play the game, meaning that their first decision is whether to play at all and continually reevaluate whether to keep playing [35], which led to studies behind why players are playing and engaged in video games. Thus, their participation is voluntary, and players expect to get an enjoyable and fun experience to keep playing [15].

Vorderer and Jennings [13] argue that most theoretical work on the enjoyment of playing video games has focused on the issue of interactivity and player action during gameplay but rarely does the game industry examine what truly engages players. Theories of "player personality" began with Bartle, speculating that players are separated based on the degree to which a player acts or interacts with other players and/or the game world creating four types of gamers (achievement for Achievers, Exploring for Explorers, Socializing for Socializers and Imposition upon others for Killers) [3]. Another model comes from Yee, who surveyed players of massively multiplayer online role-playing games (MMORPGs), identifying three factors composed of ten subcomponents [3]. These factors included: Achievement, consisting of subcomponents advancement, mechanics, and competition; Social, consisting of socializing, relationships, and teamwork; and Immersion, consisting of discovery, role-playing, customization, and escapism. Different results were showed in a survey for first person shooter (FPS) games, with gaming motives there to be more related to competition and sense of control [36]. Kahn et al. [37] focused their study on massively multiplier online (MMO) and multiplayer online battle arena (MOBA) players and their motives to play these two types of games revealing five characteristics: socializing, exploring, competing, skill development, and escapism. Li et al. [38] also studied the motives to play MMO games for adolescents and found that socializing, achievement, and immersion as important senses.

Demetrovics et al. [39] developed the motives for online gaming questionnaire (MOGQ), which included seven gaming factors: social, escape, competition, coping, skill development, fantasy, and recreation. Francisco et al. [40] in a similar study developed a videogaming motives questionnaire (VMQ), revealing eight factors to play video games: Recreation, Social Interaction, Competition, Violent reward, Cognitive development, Customization, Coping, and Fantasy. Huang and Hsieh [41] investigated the experiential factors toward online games finding significance in three motives (Entertainment, Challenge, and Control). In their research Hsiao and Tang [42], investigating factors toward mobile movie-themed games found similar results (Flow, Act, Relate). In a similar context of mobile gaming, Merikivi et al. [43] found some interesting results in relation to factors explaining enjoyment, with challenge, novelty, design esthetics, and ease of use all contributing toward making mobile games enjoyable. Finally, Wu et al. [44] found seven factors (Escape, Skill development, Fantasy, Coping, Recreation, Competition, and Social) toward Internet gaming. Lastly, report from the Entertainment Software Association (2019), most video gamers (79%) report playing video games for mental stimulation and 78% report gaming for relaxation and stress relief [7]. The above studies are summarized in the table below and showcase different results in the engagement characteristics of players depending on the type of game.

The importance of motives to play games for the development of a game has taken a new direction in the latter ages with the advanced algorithms incorporated in games. Previous studies have used traditional methodologies (such as qualitative and quantitative approaches) to understand or measure the elements that make a game fun and engaging. However, the latest technology can measure patterns in a game and measure gamers behavior and use sophisticated algorithms to replicate those tools that make a game fun for different gamers (**Table 1**).

2.3.1 Socializing

Looking at the table above reviewing the literature, it becomes apparent that the most popular influence to play games is the opportunity they create to socialize (see [14, 15, 37–40, 44]). According to Bartle [15], players would use the game as a communicative facility and apply the role-playing that these engender as a context in which to converse (and interact) with other players. Yee [14] considered Bartle's definition on socializing within games and added that players find fun in games when the social factor assesses the motivation to communicate with other players, and they form significant and long-lasting relationships with others cooperating and communicating to complete a task in the game. Kahn et al. [37] argue that games, such as *League of Legends*, capitalizing on this element adding game mechanisms and social

Author(s)	Game mode	Characteristics
Bartle	MUD	Achievement, Socializing, Exploring, Imposition upon others
Yee	MMORPG	Achievement, Socializing, Immersion
Demetrovics et al.	Motives for Online Gaming Questionnaire (MOGQ)	Socializing, Escapism, Competition, Skill development, Coping, Fantasy, Recreation
Li et al.	MMO	Socializing, Achievement, Immersion
Kahn et al.	MOBA and MMO	Socializing, Escapism, Exploring, Skill development, Competition
Laconi	FPS	Competition, Control
Francisco et al.	Videogaming Motives Questionnaire (VMQ)	Social Interaction (Socializing), Competition, Violent reward, Cognitive development, Customization, Coping, Fantasy, Recreation
Huang and Hsieh	Online games	Entertainment, Challenge, Control
Vorderer	Online games	Competition, Exploring, Achievement, Immersion, Socializing
Wu et al.	Internet games	Escapism, Skill development, Fantasy, Coping, Recreation, Competition, Socializing
Hsiao and Tang	Mobile movie-themed games (MMGs)	Flow, Act, Relate,
Merikivi et al.	Mobile games	Challenge, Novelty, Design Esthetics, Ease of use

Table 1.
Studies focused on engagement in games.

interaction protocols that will allow players to socialize and communicate. Wu et al. [44] show that in-game socializing leads to such satisfaction and strives for further building of relationships via gaming. The outcome of this taxonomy should inform the algorithms that certain groups of players are more likely to find fun in a game when allowed to interact with others and socialize for game-related or nongame-related purposes. In return, the algorithm will provide gamers with the appropriate tools to do so.

2.3.2 Achievement

Previous literature also highlights that players will find it engaging when they feel they can get a sense of achievement. These players will give themselves game-related goals and vigorously set out to achieve them [15]. According to Yee [14], the sense of achievement in games will force players to look for status of power and wealth for their in-game characters, and they will work toward understanding the game in detail to utilize the systems better than others. Li et al. [38] explain that achievement is a sense that will influence players to increase gaming time in either raiding or in player vs. player (PvP) combat. Fun through achievement occurs when players can collect points and rise in levels, and all are ultimately subservient to this [15]. This indicates that algorithms should identify players rank higher in achievement behavior and provide task to enhance that behavior.

2.3.3 Immersion: Escapism - flow

Immersion is another factor players see as fun and engaging in games. Immersion is the state where a player learns and discovers the game world, customizes their characters, creates a sense of role-play, and escapes from the daily routine [38]. Escapism is also a sense that has similar meaning as it is defined as playing to forget daily problems [40]. Vorderer [13] explains that players become actively involved in the game world and escape from their daily life when playing games leading to a sense of immersion. Looking at the definition of immersion, it is recognized that immersion and escapism are similar; hence, they are merged as one similarly to Yee [14]. Flow has a similar meaning with immersion and escapism. Demetrovic et al. [39] define flow as the state of concentration or complete absorption with the game, leading to a sense of engagement that strong during which temporal concerns (time, food, ego–self) are typically ignored. Previous studies identified that games allow players to escape from their daily routine and transfer into an imaginary world, enhancing the element of fun.

2.3.4 Competition

Competition is another important sense in game enjoyment. It is similar in meaning with achievement; however, it occurs through the desire of social comparison [38]. For example, Demetrovic et al. [39] find that competition requires some form of comparison of achievements since players like to "...defeat others" or "... compare knowledge with others". Wu et al. [44] agree with the social comparison promoted by the sense of competition defining it as competing with others. Looking at the discussion of around competition, it becomes apparent that even though there is a feeling of achieving for the player, it is not enough to enhance the element fun, but it requires some comparison with other players in the game making it interactive with others. Bartle [15] agrees that there should be a division between the two categories saying that some people find element of fun when they are imposing upon others and killing off their personae (characters) with joy coming from the greater the distressed caused. Algorithms in games should be able to identify the difference in behaviors and cluster players with higher levels of competitiveness separated from gamers with higher levels of achievement. This is translated to activities that enhance competitiveness between gamers and more interaction with others than the system.

2.3.5 Exploring

Some of the reviewed literature highlights that players will find it fun when they can explore the virtual world. Bartle [15] was the first to express that highlighting that this group of players find fun having the game expose its internal machinations to them, by mapping its topology, to then advances to experimentation with its physics. Bartle's statement recognizes the interaction between the gamer and the system and the behavior to exploit hidden components in the environment. This interaction should help algorithms measure the behavior and provide tasks to enhance the behavior. Yee [14] agrees with this opinion mentioning that players find it fun when they can discover items or find hidden things in a map. Kahn et al. [37] further discusses players exploration activities in the virtual world identifying that these players like to explore hidden areas in the game and the game mechanics to optimize their game performance. These groups of players like to explore and conquest unknown scenes and possibilities [39]. It is then important for these players to interact with virtual worlds and discover hidden treasures.

2.3.6 Customization

Another very important element contributing toward the meaning of fun in games is the ability they provide to the player to customize. Kim et al. [45] define customization as the activities, where users can modify some aspect of an interface to a certain degree to increase its personal relevance. Shaw et al. [46] mention that several researchers have already looked at the relationship between avatar customization and player enjoyment of games. Kim et al. [45] agree with this opinion and explain that customization is a very critical sense to enjoyment. Customization will include any color changes to game elements, choice of background pictures, avatar customization but it may also include selection of background music sounds, and other features [47]. Shaw et al. [46] do add to this opinion finding that in the context of games, players ability to customize characters leads to greater feelings of presence and psychophysiological indicators of emotion. The ability of games to allow gamers to customize their experience is what differentiates the game industry with other entertainment industries. It could then be argued that when players play games will most likely be interested in building the character itself rather than the story or the level of the game. Even though customization has been directly linked with immersion [14] as it allows players to build an avatar and become actively involved in a virtual world, is seen as independent meaning of fun as it does not immediately result in that outcome. Instead, the sense of immersion could be a result of other mechanics of the game.

2.3.7 Challenge: Skill development

Literature also explains that players find fun in skill development and challenges. Meriviki et al. mention that players seek challenges, and for a game to be challenging, it must provide personally meaningful in-game goals whose attainment is not certain [43]. For instance, in first person shooter (FPS) games competition and challenge scored higher than other dimensions for committed players [37]. Among other things, challenges are important motivations for gameplay as they are interactional dynamics between players and videogame system [48]. Even though the meaning of achievement, competition and challenges look similar they have a main difference since competition is compared and interactive with other players when achievement is interactive with the system. Challenges in a game can be interactive with other players or personal, hence they are independent typologies in the meaning of fun. Instead, challenges are more related in meaning with the skill development theme described by Wu et al. Demetrovics et al. and Kahn et al. Hsiao and Tang [42] argue that for engagement to occur both challenges and skill development must exist and exceed the level of difficulty that is typical for the individual's day-to-day experiences. This shows that players find fun when the game promotes a challenge, where they must come up with a strategy and improve their skill to overcome the tasks and level up.

2.3.8 Esthetics

Design esthetics seems to also contribute toward creating a fun environment for players. Merikivi et al. [43] defined design esthetics as the degree to which a person believes that the platform is esthetically pleasing to the eye. It can then be argued that good design esthetics, such as music, sound effects, and animations, are important components of a video game experience and can contribute toward tolerating other imperfections in game design. Atkinson and Parsayi [49] do mention that game mechanics is a complex context, and every aspect contributes toward enhancing the element of fun for the player experience. For example, plot animations and pictures, which are used to reward important events (such as defeating an enemy or leveling up) contribute toward making that event more fun. Cao [50] discussed the importance of esthetics in video games highlighting the importance of visual art and esthetics in the progress of the game.

2.4 The concept of fun

In previous sections, the overview of games showed their success as an industry. The concept of game in relation with algorithms was examined, and studies that explore gamers' preferences were explored, identifying eight main elements (Socializing, Achievement, Immersion or Escapism or Flow, Competition, Exploring, Customization, Challenge or Skill development, and Esthetics). Games looking into applying elements to enhance the sense of fun should be aware that players will show preference in at least one or more of these senses. The acknowledgment of this typology of players leads to valuable practical implications as it provides a useful tool for designers to understand their audience to classify players' profiles and promote mechanics that may be more likely to enhance the sense of fun leading to engagement with the game. Game developers should investigate mechanics that can enhance fun since players are likely to be engaged to use the system because of these elements. The outcome of this engagement in addition to the evolved algorithms will create a highly personalized content promoting individual preferences and creating an attractive environment. The classification suggests that game designers should focus on building an experience in the game that satisfies all eight typologies, so to create a desirable ecosystem, where every player will find something fun to stay in the game for.

Previous studies investigated the characteristics of players when playing games with a variety of results depending on the game. Bartle [15] identifies a common characteristic behind those studies clarifying that in the topic of games the answer to the question of "Why do people play?" sounds obvious at a superficial level since people are playing games for fun, but what creates the fun? Fun is a subjective term, and it could result in so many different features that must go into a game to make it fun for everyone. Despite, the significance of their results that question remained open. This chapter is trying to answer this question exploring eight characteristics of the meaning of fun reviewing literature. The biggest outcome of the literature is that a game should be balanced for all players with every typology helping the system to thrive. Looking at the results of the table it becomes apparent that different games serve different enjoyment, hence different fun elements. Bartle highlights that, clarifying that in his research every player used to tend at least a bit at all four elements, however, they had a bigger tendency toward one. For example, players who explain fun as achievement still enjoyed socializing, exploring, and imposition upon others if it could lead them to more tasks to achieve [15]. This is to highlight the complexity of the term fun in the game since each typology is not independent, but it relies on every other one to enhance the sense of fun in the game. Taking that into consideration, it is assumed that every player will have a tendency at least a bit at all eight elements, however, they will have a preference toward one (see **Figure 1**). The diagram measures fun on a scale from 1to 10, depending on the behavior of a player in-game activities and engagement to tasks and mechanics.

For this example (**Figure 1**), the game includes mechanics to promote all eight characteristics and attract every player. The player shows engagement in mechanics and tasks that promote socializing, and the behavior is tracked through algorithms to match the profile of someone with a social preference in the meaning of fun. However, the diagram shows that the player still engages in activities that promote other behavior but on a smaller scale. In this essence, the player will find the mechanics that will make the game fun mostly through socializing; however, they

Figure 1.
Socializing in a perfect environment.

will be looking into other tasks in the game since they might help them to get more opportunities to socialize. The same player in the second game (**Figure 2**) does not find enough game mechanics promoting socializing; therefore, it is more likely that the game is not that fun and will look at other games. Even though, the player will fit into the game and engage with mechanics that promote fun through achievement, competition, or challenge. Still, the fact that the game does not promote socializing enough would probably make the game less fun for the player. The example is to show that each player in a game is unique, and they have unique interpretation of the meaning of fun. This meaning of fun can be measured through the behavior of a player in-game activities and engagement to tasks and mechanics. This would then create a psychographic for the individual, which will then classify what mechanics will be more attractive for the player. Every player in a game will fall into a similar but independent diagram because every player will be different.

It is important to highlight that beside the personal psychograph for the player, there is also the relationship between players that might affect the perception of fun for each one. For example, (**Figure 3**), when a player high in preference in achieving meets a player with high preference in socializing could affect the fun elements for both. In the diagram below, it is shown how the player high in socializing is now ranked higher (from 4 to 6) in the achievement element since they both enjoy playing together. This is to show the complexity of measuring the element of fun for players when they play a game. However, it is important for a game to understand the behavior of each player and the mechanics that make the game more attractive for the player to enhance the engagement between the player and the game.

As video games exhibit progressively expansive game environments, there has been growing interest in employing generative computational algorithms to mitigate the cost of authoring game content [27]. These algorithms can formulate these diagrams helping the game to provide the appropriate tasks for each player and make the experience more personal and engaging. As Vorderer and Jennings [13], discussed the main tool enhancing the engagement in games and differentiating them with

Figure 2.
Socializing in a lacking environment.

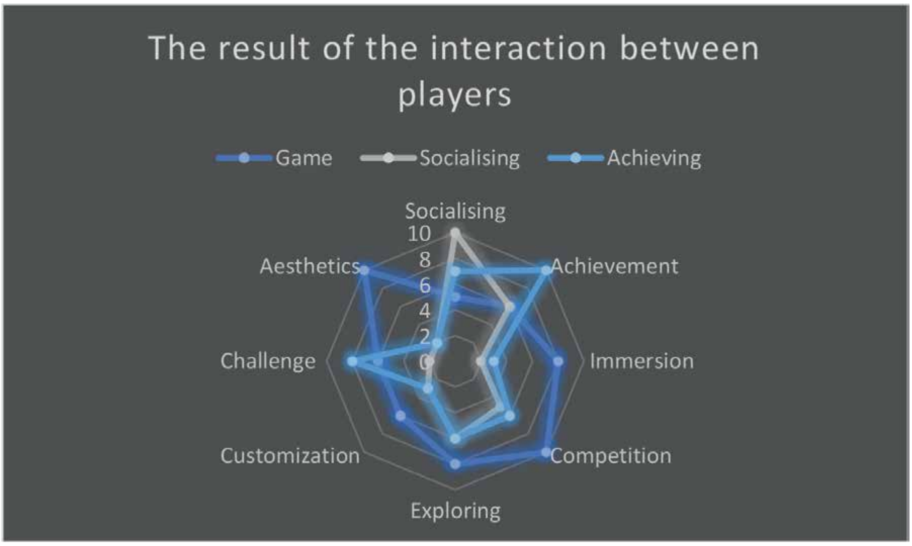


Figure 3.
Interaction between players.

other entertainment industries (such as movies) is the element of control that games provide to the player. This is the result of advanced algorithms creating opportunities for the game to track previous behavior and provide appropriate tasks and mechanics, so the player has the desired tools to progress in the game. As a result of this interaction, the game can measure the element of fun for each player and enhance this feeling, by providing the appropriate tools.

3. Conclusion

Professional game designers often take it for granted that people just want to play their games, rarely examining the psychology of their gaming audience [13]. Previous studies (see [6, 13, 14]), investigated the behaviors of players when playing games creating a typology of them. However, previous literature with limited exceptions (see [15]) has not investigated in depth the meaning of fun for the players ecosystem. Hence, this chapter focused on exploring the meaning of fun in games, which results in increased interaction with the gamer and absorbing more data to feed the algorithms. The idea is that the more data an algorithm absorbs the more advanced experience it can create for the gamer. In today's game science, it is very important for a game developer to apply sophisticated algorithms, which in return will create a fun experience for the gamer not only to play the game but also return to the environment frequently. Personal data collected can be a tool for game designers to develop future games for specific audiences.

To satisfy this aim, this chapter has reviewed gaming literature and previous research that focused on the behaviors of players in games and their preferences in several types of games. Using some of these findings, this chapter created a diagram to measure players' behavior in a game. From there, it suggests that players' meaning of fun can be measured by ranking previous behavior to explain the fun element for each player. Literature review revealed that the meaning of fun is subjective for each player, and it is not specific to one characteristic but a combination of a few senses. This chapter recognized eight senses from the literature (Socializing, Achievement, Immersion or Escapism or Flow, Competition, Exploring, Customization, Challenge or Skill development, and Esthetics) a game can use to enhance the feeling of fun for players. Each player will find fun in each of these eight senses, however, there will be a significant tendency toward one of them. It is suggested that every player in a game will develop such a psychograph based on behavior and patterns.

The outcome of this chapter is the presentation of diagrams showcasing the relationship between fun and gaming algorithms. Presenting an example of a player with tendency toward socializing, it showcases that a player will find the mechanics that will make a game fun mostly through socializing mechanics, and other mechanics in the game, they will be ranked supplementary in importance. This is to highlight the importance of including enough social mechanics for that player to enhance the meaning of fun; otherwise, it is likely that the player will not find the game as fun as others and lose interest and engagement. Considering the nature of the topic, it is also presented an example where interaction between the social player with other players (in this case someone ranked high in achievement) will affect the previous behavior and as a result, will affect the overall fun aspects of the game deteriorating the diagram.

The concept of fun is difficult to describe, especially within a game, since it offers a special intrinsic satisfaction to the player, which will eventually lead to the engagement with the game and play. Recreating fun is expenditure of time and technology in a manner designed for therapeutic refreshment of an individual's body or mind in the form of play activity. The relationship between play and fun in games is found to be very complex and dependent on the way it is framed. Schell [51] tried to put a definition on play referring to the term as activities, which are accompanied by a state of comparative pleasure, exhilaration, power and the feeling of self-initiative, and other characteristics such as imagination, competition, and problem-solving. This chapter tried to explain the previous statement, by creating the fun measuring diagram. It is believed that game designers should acknowledge the meaning of fun for players when building a game and include algorithms to measure players behaviors in a game to promote the appropriate tools so to create the desired engagement.

Author details

Demos Parapanos
Institute of Business, Industry and Leadership, University of Cumbria, Carlisle, UK

*Address all correspondence to: demos.parapanos@cumbria.ac.uk

References

[1] Bowman ND, Kowert R, Cohen E. When the ball stops, the fun stops too: The impact of social inclusion on video game enjoyment. Computers in Human Behavior. 2015;**53**:131-139. DOI: 10.1016/j.chb.2015.06.036

[2] Sajid MJ, Cao Q, Brohi A, Sajid M. Video gaming a new face of inducement tourism: Main attractors for juvenile gamers. International Journal for Social Studies. 2018;**4**(5):52-56

[3] Hilgard J, Engelhardt CR, Bartholow BD. Individual differences in motives, preferences, and pathology in video games: The gaming attitudes, motives, and experiences scales (GAMES). Frontiers in Psychology. 2013;**4**:608. DOI: 10.3389/fpsyg.2013.00608

[4] Cox J. What makes a blockbuster video game? An Empirical Analysis of US Sales Data. Managerial and Decision Economics. 2014;**35**:189-198. DOI: 10.1002/mde.2608

[5] Costes J-M, Bonnaire C. Spending money in free-to-play games: Sociodemographic characteristics, motives, impulsivity and internet gaming disorder specificities. International Journal of Environmental Research and Public Health. 2022;**19**:15709. DOI: 10.3390/ijerph192315709

[6] Michelini Y, Ibáñez MI, Pilatti A, Bravo AJ, López-Fernández FJ, Ortet G, et al. Motives to play videogames across seven countries: Measurement invariance of the Videogaming motives questionnaire. Addictive Behaviors. 2023;**140**:107624. DOI: 10.1016/j.addbeh.2023.107624

[7] de Hesselle LC, Rozgonjuk D, Sindermann C, Pontes HM, Montag C. The associations between big five personality traits, gaming motives, and self-reported time spent gaming. Personality and Individual Differences. 2021;**171**:110483. DOI: 10.1016/j.paid.2020.110483

[8] Xu F, Tian F, Buhalis D, Weber J, Zhang H. Tourists as Mobile gamers: Gamification for tourism marketing. Journal of Travel & Tourism Marketing. 2016;**33**:1124-1142. DOI: 10.1080/10548408.2015.1093999

[9] Chen L, Zhang P, Li S, Turner SF. Growing pains: The effect of generational product innovation on mobile games performance. Strategic Management Journal. 2022;**43**:792-821. DOI: 10.1002/smj.3338

[10] da Silva Brito RD, Contreras Pinochet LH, Luiz Lopes E, de Oliveira MA. Development of a gamification characteristics measurement scale for mobile application users. Internext. 2018;**13**:1-16. DOI: 10.18568/1980-4865.1311-16

[11] Rapp A. Social game elements in world of Warcraft: Interpersonal relations, groups, and organizations for gamification design. International Journal of Human-Computer Interaction. 2018;**34**:759-773. DOI: 10.1080/10447318.2018.1461760

[12] Parapanos D, Michopoulou E. Innovative Mobile Technology in Hotels and the use of gamification. Tourism Planning & Development. 2022;**20**(2)162-187. DOI: 10.1080/21568316.2022.2107563

[13] Vorderer P, Bryant J. Playing Video Games. London: Taylor and Francis; 2012. DOI: 10.4324/9780203873700

[14] Yee N. Motivations for play in online games. CyberPsychology & Behavior. 2006;**9**:772-775. DOI: 10.1089/cpb.2006.9.772

[15] Bartle RA. Designing virtual worlds. Berkeley, Calif: New Riders; 2004

[16] McDaniel PA, Forsyth SR. Exploiting the "video game craze": A case study of the tobacco industry's use of video games as a marketing tool. PloS One. 2019;**14**:e0220407. DOI: 10.1371/journal.pone.0220407

[17] Smith LJ, Gradisar M, King DL. Parental influences on adolescent video game play: A study of accessibility, rules, limit setting, monitoring, and Cybersafety. Journal of Child and Family Studies. 2015;**18**:273-279. DOI: 10.1089/cyber.2014.0611

[18] Adachi PJC, Willoughby T. Does playing sports video games predict increased involvement in real-life sports over several years among older adolescents and emerging adults?. Journal of Youth and Adolescence. 2016;**45**:391-401. DOI: 10.1007/s10964-015-0312-2

[19] Rogers R. The motivational pull of video game feedback, rules, and social interaction: Another self-determination theory approach. Computers in Human Behavior. 2017;**73**:446-450. DOI: 10.1016/j.chb.2017.03.048

[20] Hemovich V. It does matter if you win or lose, and how you play the (video) game. Games and Culture. 2021;**16**:481-493. DOI: 10.1177/1555412020913760

[21] Chan ESW, Okumus F, Chan W. The applications of environmental Technologies in Hotels. Journal of Hospitality Marketing & Management. 2017;**26**:23-47. DOI: 10.1080/19368623.2016.1176975

[22] Bigl B, Schlegelmilch C. Are video games still a boys' Club? How German public television covers video games. Games and Culture. 2021;**16**:798-819. DOI: 10.1177/1555412020975637

[23] Pappas IO, Mikalef P, Giannakos MN, Kourouthanassis PE. Explaining user experience in mobile gaming applications: An fsQCA approach. Internet Research. 2019;**29**:293-314. DOI: 10.1108/IntR-12-2017-0479

[24] Mulyawan I, Rafdinal W. Mobile games adoption: An extension of technology acceptance model and theory of reasoned action. IOP Conference Series. Materials Science and Engineering. 2021;**1098**:32022. DOI: 10.1088/1757-899X/1098/3/032022

[25] Yu RWL, Chan AHS. Meta-analysis of the effects of game types and devices on older adults-video game interaction: Implications for video game training on cognition. Applied Ergonomics 2021;**96**:103477. DOI: 10.1016/j.apergo.2021.103477

[26] Schenk S, Lech RK, Suchan B. Games people play: How video games improve probabilistic learning. Behavioural Brain Research. 2017;**335**:208-214. DOI: 10.1016/j.bbr.2017.08.027

[27] Sorenson N, Pasquier P, DiPaola S. A generic approach to challenge Modeling for the procedural creation of video game levels: IEEE Transactions on Computational Intelligence and AI in Games. 2011;**3**:229-244. DOI: 10.1109/TCIAIG.2011.2161310

[28] Hoffman B, Nadelson L. Motivational engagement and video gaming: A mixed methods study. Educational Technology Research and Development. 2010;**58**:245-270. DOI: 10.1007/s11423-009-9134-9

[29] Rafiq A, Asmawaty Abdul Kadir T, Normaziah Ihsan S. Pathfinding

algorithms in game development. IOP Conference Series. Materials Science and Engineering. 2020;**769**:12021. DOI: 10.1088/1757-899X/769/1/012021

[30] Parapanos D, Michopoulou E. Meaning of fun in hotel gamified applications. In: Information and Communication Technologies in Tourism 2023, (ENTER, Johannesburg, Springer) 2023;**15**. DOI: 10.1007/978-3-031-25752-0_2

[31] Siemens JC, Smith S, Fisher D, Thyroff A, Killian G. Level up! The role of Progress feedback type for encouraging intrinsic motivation and positive brand attitudes in public versus private gaming contexts. 2015;**32**:1-12. DOI: 10.1016/j.intmar.2015.07.001

[32] Andrezej M. Marczewski's Gamification User Types 2014. https://elearningindustry.com/marczewski-gamification-user-types

[33] Melodia F, Canale N, Griffiths MD. The role of avoidance coping and escape motives in problematic online gaming: A systematic literature review. International Journal of Mental Health and Addiction 2022;**20**:996-1022. DOI: 10.1007/s11469-020-00422-w

[34] Parapanos D, Michopoulou E, (Elina). Let's all Play Together: Motivations of Different Gamification User Types. Hershey, PA, USA: IGI Global; 2022. DOI: 10.4018/978-1-7998-9223-6.ch001

[35] Morgan McGuire OCJ. Creating Games. Florida: CRC Press; 2008

[36] Laconi S, Pirès S, Chabrol H. Internet gaming disorder, motives, game genres and psychopathology. Computers in Human Behavior. 2017;**75**:652-659. DOI: 10.1016/j.chb.2017.06.012

[37] Kahn AS, Shen C, Lu L, Ratan RA, Coary S, Hou J, et al. The Trojan player typology: A cross-genre, cross-cultural, behaviorally validated scale of video game play motivations. Computers in Human Behavior. 2015;**49**:354-361. DOI: 10.1016/j.chb.2015.03.018

[38] Li DD, Liau AK, Gentile DA, Khoo A, Cheong WD. Construct and predictive validity of a brief MMO player motivation scale: Cross-sectional and longitudinal evidence based on Singaporean young gamers. Journal of Children and Media. 2013;7:287-306. DOI: 10.1080/17482798.2012.712918

[39] Demetrovics Z, Urbán R, Nagygyörgy K, Farkas J, Zilahy D, Mervó B, et al. Why do you play? The development of the motives for online gaming questionnaire (MOGQ). Behavior Research Methods. 2011;**43**:814-825. DOI: 10.3758/s13428-011-0091-y

[40] Francisco,J. L-F, Mezquita L, Griffiths MD, Ortet G, Manuel I.I. The development and validation of the Videogaming motives questionnaire (VMQ). PloS One. 2020;15:e0240726. DOI: 10.1371/journal.pone.0240726.

[41] Huang L-Y, Hsieh Y-J. Predicting online game loyalty based on need gratification and experiential motives. Internet Research. 2011;**21**:581-598. DOI: 10.1108/10662241111176380

[42] Hsiao C-H, Tang K-Y. On the post-acceptance of mobile movie-themed games. Electronic Commerce Research and Applications. 2016;**18**:48-57. DOI: 10.1016/j.elerap.2016.06.003

[43] Merikivi J, Tuunainen V, Nguyen D. What makes continued mobile gaming enjoyable?. Computers in Human Behavior 2017;**68**:411-421. DOI: 10.1016/j.chb.2016.11.070

[44] Wu AMS, Lai MHC, Lau JTF, Lei M-W, Yu S. Motives for online gaming questionnaire: Its psychometric properties and correlation with internet gaming disorder symptoms among Chinese people. Journal of Behavioral Addictions. 2017;**6**:11-20. DOI: 10.1556/2006.6.2017.007

[45] Kim K, Schmierbach MG, Bellur S(S), Chung M-Y, Fraustino JD, Dardis F, et al. Is it a sense of autonomy, control, or attachment? Exploring the effects of in-game customization on game enjoyment. Computers in Human Behavior. 2015;**48**:695-705. DOI: 10.1016/j.chb.2015.02.011

[46] Shaw A, Kenski K, Stromer-Galley J, Mikeal Martey R, Clegg BA, Lewis JE, et al. Serious efforts at Bias reduction: The effects of digital games and avatar customization on three cognitive biases. Journal of Media Psychology. 2018;**30**:16-28. DOI: 10.1027/1864-1105/a000174

[47] Javora O, Děchtěrenko F, Tetourová T, Volná K, Brom C. Customization in educational computer games and its effect on learning: Experimental study with primary school children. Journal of Computer Assisted Learning. 2021;**37**:1370-1382. DOI: 10.1111/jcal.12576

[48] Vahlo J, Karhulahti V-M. Challenge types in gaming validation of video game challenge inventory (CHA). International Journal of Human-Computer Studies. 2020;**143**:102473. DOI: 10.1016/j.ijhcs.2020.102473

[49] Atkinson P, Parsayi F. Video games and aesthetic contemplation. Games and Culture. 2021;**16**:519-537. DOI: 10.1177/1555412020914726

[50] Cao J. The collector, the Glitcher, and the Denkbilder: Toward a critical aesthetic theory of video games. Games and Culture. 2023;**18**:322-338. DOI: 10.1177/15554120221097411

[51] Schell J. The Art of Game Design : A Book of Lenses. Boca Raton: CRC Press; 2015

Chapter 3

Towards Standardizing Game Designing Processes: Game Designers' Initiation Tool

Mifrah Ahmad

Abstract

This chapter intends to present game designers' perspectives which elaborates on the ways they approach designing games. While it is diverse, the interpretive analysis of 17 game designers' perspectives will be able to shed light on how game-designing processes may be influenced by their experience-based and reflective practice, or whether they apply formal methods based on how game-designing fields represent. Furthermore, it discusses whether the formal methods explained by the game designers resonate with existing literature and compare the relevance between the practices and the literature that advocates best practices. The results are discussed through interpretivism paradigm, to provide the essence of game designers' experiences and approaches. Lastly, it concludes with a list of common practices in their designing processes for games and a future direction towards how we can understand game designers' perspectives and ease the designing process of games.

Keywords: game design, designing process, experience-based design, experiential learning, interpretivism, interpretive analysis, phenomenology, reflective practice, formal methods

1. Introduction

The Australian game industry has experienced a boom over the past few years. As reported by Interactive Games and Entertainment Association (IGEA), 76% of players are actively playing video games [1]. Parents reason that playing video games is a way to help educate their children (19%), and 79% of them generally use games for education [1]. Lastly, the survey indicated that 86% of parents play games that are relevant to general knowledge, 83% games provide digital knowledge, 71% benefit gameplay by gaining life skills, and 74% suggested that games provide cultural knowledge [1]. Video games are reportedly used in schools for five reasons: teach students (71%), motivate them (70%), help them pay attention (64%), help teachers teach (63%), and help schools remain relevant (62%) [1]. While this presented interesting facts about how games are used in learning, the effectiveness of using games was reasoned as follows: gaining new knowledge (34%), new software/tools (30%), and new skills (26%) [1]. The survey from IGEA 2020 revealed that the games industry grew by 29% year-on-year in 2020 and is now at a net worth of $185 million [2]. The development of Australian video game platforms has dramatically increased along with challenges faced in the industry, including a lack of international events, limited contact with investors, reduced productivity, and a decrease of contractual revenue [3, 4]. While recommendations were made to restore the Australian Interactive Games Fund to support game developers [5], similar challenges were listed in the 2021 report with an added challenge of "lack of appropriate government funding", "attracting early-stage development funding", and "hiring employees with specialized skills" [6]. The preferred platforms for game development were PC (48%) and mobile/tablet (44%). Hence, the digital games or video games used and discussed within this research project and participants are playable on PC and iPad.

Games have the potential to provide an experience that a player encounters through a series of interactions with the system. Costikyan [7] suggests that a game is a form of art in which participants, terms players, make decisions in order to manage resources through game decisions in the pursuit of a goal. Games have existed for decades and have shown a positive influence in educational settings. Here, they are intended to promote the development of cognitive skills or to practice various skills in a virtual environment [8]. Defining games relies upon various aspects and points of view. Stenros [9] reviewed 60 game definitions clearly showed the difficulty of gaining consensus on game design requirements, and recommended the need to avoid building limitations by adopting ontological foundations. Prominently, gaming research intends to define a more general game definition as, "a system in which players engage in artificial conflict, defined by rules, that results in a quantifiable outcome" [10].

To examine and simplify educational game (EG) design process, it is crucial to address general game design process first, because it predates specific notions such as EG [11, 12]. Recent studies have shown that the EG design process involves various stakeholders, with interdisciplinary collaboration necessary to move towards providing successful EGs for schools [13, 14]. It asserts the need for an in-depth and empirical analysis of game developers, video game designers, and teacher perspectives to explore collaborative research design approaches. Moreover, an interdisciplinary debate about whether teachers' experiences may inform game designers' decision-making processes has recently been addressed as a gap, by several authors [15–17]. Recently, Keogh has reported that game designers' perspectives require theoretical and empirical attention to understand a "*fuller range of dispositions held by videogame makers*" [13, 14]. This requires a focus on how game designers differentiate between the approaches of designing games and EGs processes, and whether their inspirations, desires, and beliefs influence such practices. It is useful to consider game designers and teachers first as games/EG play testers, who construct their experience as they play and design. Such constructed experiences require an empirical and theoretical unpacking of their experiences and perspectives on the phenomenon.

2. Related literature

Game design is known for its complex nature, which includes multiple iterations and crafting work. Game design is known for its artistic, creative processes that require technical integration [18]. Zimmerman [19] described it as an iterative designing process where designers craft play and construct a set of rules directing players' play experience, while indirectly connecting game system rules to enable this experience. From a developer's perspective, iteration can be perceived as polishing and improving a product to ensure its requirements are functional [20]. The nature of the iterative design process is such that those designed rules are inhabited and enacted by the players, which in turn "creates an emerging pattern of behaviour, sensation, social exchange, and meaning" [19]. The intention of game design is to address its compelling nature through play. Game software design has been increasingly used to engage with players/students, enabling players to develop skills and knowledge to succeed in the digital world [21]. Cross [21] states that design knowledge is deeply rooted in the various processes and people involved during the process. The techniques and strategies designers use guide them in their work; the built products are also held in such a way that shows how designed artifacts should be made.

As game design processes derive from software design and development; design, art, software, and teams all work together on developing processes to produce a finished product [22], and it is perceived to achieve positive results [23, 24]. Prominent game design guides devote whole chapters to discussing the processes or stages of designing games [11, 25–27]. For instance, Fullerton [27] frames the core concepts of designing games through a play-centric model, which is iterative (cyclic) in nature and repeats the phases (idea generation, conceptualization, testing, and evaluation) to advocate for the needs of the players. Schell [11] elaborates on the importance of game experience and elements while surrounding the discussion on tools and approaches with self-reflective, iterative, and game designers' experience. For Schell [11], the lack of systematic view or transparency has raised some debates in academia. It is not theoretically organized, but more relevant to practicing design [28]. Salen and Zimmerman [10] also emphasized the testing and iterations of the game as a product to ensure it provides a meaningful experience for the player. Similarly, Adams (2014) emphasizes iterative refinement to elaborate on the design by implementing game components and features and refining them through testing.

Game development also include agile methods, which have been reported to be beneficial, based on placing an iteration at the beginning of the designing process [29, 30] or using iterative design through a scrum (see details of agile[1] method [29] to enable sprints (short documentation of progress through designing and planning game designing procedures) for a reliable end product. To briefly explain, software development methods such as SCRUM, Lean, or Kanban are adapted and applied in game design and development due to games being software [30, 31] and that games are potentially improved with iterations of each prototype [32–34]. Larman's meta-analysis also suggested models, including spiral[2] models for their cyclic nature, or risk-driven approaches rather than strictly specification-driven or prototype-driven processes [32, 34, 36]. Lastly, game design using software modeling approaches and emphasizing the iterative/cyclic nature of designing games requires more empirical or in-depth research methods, that involve the voices of game designers and developers who can share their daily practices in the field [20]. As Kultima [20] suggests, the nature of game development and design is "volatile" and tends to have limited time to form an explicit methodological approach. This does not mean that game developers and designers are not applying/adapting to existing models. Rather, there are various

---

[1] Agile software development method includes processes: requirement, Design and Develop, Test (cyclic), and deploy [30].

[2] Spiral model includes processes: determine objectives, identify and resolve risks, development and testing and plan the next iteration (cyclic) [35].

assumptions, design values, beliefs, and prior software design experiences that potentially guide their thought processes. As developers iterate, reflect, and adapt on past successes or failures, these assumptions and values may not always be immediately apparent to them [33, 35, 37, 38].

Due to the adaptive nature of game design, no explicit models are suggested that produce the desired end product, as games are a multidisciplinary field of research that positions the players/intended audiences as end-users [28, 39, 40]. Kultima [28] also argued that game making is not a part of basic education; hence, it has the potential to be misunderstood. It is because game design praxeology sheds light on the experiences that creators go through while making, designing, and developing such games. With game design processes being pluralistic, opportunistic, and iterative [28], their timeliness and the values integrated within the designing are considered a pool of ideas that are combined within a system to create such experience. The published literature acknowledges that a new game can be approached from many different points of view (concept, art, theme, object, etc) [11, 26, 39]. One way to initiate a concise and narrower approach to designing a game is to focus on the game's core elements [39]. Core elements include rules, core mechanics, the setting or the environment, and the characters or story [39, 41]. This approach is called the game as an artifact.

On a different note, the game development and designing process is centred around 'who are the audience or the players of the game'. This is because it influenced the decision-making process of selecting designing core elements with the notion of player experience as a priority [11, 25, 42]. Hagen [39] called this approach designing for player experience. Hagen [39] posited that Dewey's art as experience [43] is relevant to user experience, and advocated for using function (usability of the game) and experience (look and feel) in game design. Hagen [39] reported autobiographical responses of a game design team (game designer, developer, lead designer, etc.) and suggested the need to encourage game designers to reflect upon their experiences and practices or employ different approaches that could be insightful in gaming industry. However, Hagen acknowledges that the experiences of game designers are subjective, and that interviewing one design/development team does not provide rigorousness in understanding the design experience.

In relevance with experience of game designers and players, an expanded game experience (EGE) model was proposed to consider a wide range of factors [44]. The EGE model provides a detailed description of stages and game states that game designers can consider when making conflicting design choices, however, it is yet to be applied by game designers in practice. The factors include:

a. Whether the focus of the game is relevant to a formal structure,

b. The experience required through play,

c. The continuum of gameplay experience.

As described above, a conventional software design process is driven by a list of requirements or specifications; however, in most cases, it is driven by the user's experience [11, 40]. Here, game designers' position themselves in the players' minds where they anticipate what the latter will experience, and then consider their decision-making by designing the game with compelling experiences, similar to achieving desired play experience [40]. Moreover, this is also considered experience-based communication, which seeks to "impart knowledge or skills to users," as it is crucial for the game to influence players' personal beliefs and attitudes [40]. Another study presented a Simple Model of a Game (SMG) and explained the design process in the following manner [45]:

a. Idea generation: game goals (effects on player emotions), topics (setting/world/environment), scope, worthiness, feasibility, and features,

b. Design process: A design document (detailed steps of game communication and interaction between players and the game), the game structure (how does the system work?), the scope and its interaction with the player while playing, interactivity style and implementation details,

c. Implementation/development,

d. Testing/modification in relevance to game goals.

Zagal et al. [45] advocated that the success or failure of a game "ultimately depends on the underlying ideas and skills of the designer and developers" (p. 450). Although the model is informative and consistent with software design processes such as agile or scrum, it suggests the need to involve the views and skills of game designers to explore the unique human qualities required.

In addition to being a commercial endeavor, game creation also involves a process that goes beyond just satisfying customers. This complexity adds to the generalizations of the design process in game research. For example, a study observed game design literature and the notion of user-centered design and introduced the concept of "abusive game design", which explains the challenges of design practice in relation to player advocacy [46]. Drawing such generalization within game design research can potentially add complexity [28]. Another study reported three overlapping phases in design research [47]:

1. Experiential (designers draw from their personal experiences of the design processes)

2. Intellectual (formation of theoretical constructs for design methodologies, principles, and methods)

3. Experimental/empirical (consideration of empirical published research on current practices)

Kultima [28] stated that positioning such theoretical progress aligns with the current field of game design research; however, it is a generic approach. In addition, in the game design literature, the term design patterns is relevant to all game developers [48, 49], however, it has been reported that it is not as valuable in daily practice [28]. The design patterns refer to game designers' patterns for making games, and how this shapes what they create. That is, design work is more solution-oriented than problem-oriented and uses the designers' own preceding reference systems.

Moreover, the playability and design patterns in EG have been reported to enhance player experiences that include interaction and blending of educational objectives; however, the tools and guidelines suggested need further explanation by game designers in their practice, as the proposed playability development processes requires heuristics from various stakeholders to test with players. Here, it is essential to note that the design pattern alone is not the solution to a successful game-designing process; however, they are part of game designers' choices when doing so.

2.1 Magical powers of game designers

The initial assumption about game designers is that they approach learning and teaching through games differently; there is a need to understand and organize the data from game designers' interviews using a different framework. An EG design is challenging for game designers as they have to correlate, create, and reduce complexity, but facilitate learning. As Kalmpourtzis ([12], p. 53) quotes, "*Various factors correlate with each other, creating a complex, yet magical, opportunity to facilitate learning*". The game-designing process has been repeatedly mentioned as an iterative task that is reflected upon with the game's primary goal in mind. With this thought, "*We could try to think of educational game design as a magical triangle, at each corner of which is an important aspect of the design process: Players, Game, and Learning Aspect*" ([12], p. 53). This triangle represents "the Magical Powers of the Game Designer", as a game, that includes educational and fun elements needed to make use of all three aspects. In addition, the game designer is the center of the triangle, and all aspects interact and are inseparable; as the quote suggests, "*The role of an educational game designer in this sense is to bring balance to this triangle in order to create an engaging learning game*" ([12], p. 55).

Alongside this magical triangle, there are four 'filters' that affect the way games are developed. They highly influence the game designers' perspectives: (1) The Filter of your Inner self, (2) The Filter of Players' Expectations, (3) The Filter of Designers' Ethos, and (4) The Filter of Teaching Accuracy.

Referring to the Magical Triangle ([12], p. 54), players, game, and learning aspects and their inter-relationships become essential elements to create games such as story, mechanics, and their interactions [13]. The emphasis is on game designers to ensure that the balance is maintained in the system between the realistic and imaginative context of design. But it is not clear how all these filters fit into the Magic Triangle (**Figure 1**). Therefore, the need to empirically explore the perspectives, beliefs, and values of game designers that could potentially influence their decision-making through design processes has not yet been theoretically examined.

3. Theoretical and conceptual relevance of data analysis and interpretation

The conceptual and theoretical framework adopted in this paper relates to Dewey's theory of experience [50] concepts: growth of experience, purpose, and continuity. As these concepts were relevant to the epistemology and ontology of the research, constructivism belief is situated as the paradigm to allow the construction of new knowledge which welcomes the subjectivity of individual experiences [50]. The growth of experience represents how individuals grow in their careers, experience everyday activities to gain knowledge, learn to achieve their goals, sustain their understanding of the phenomenon, and continue to grow in their experience. The aspects of 'growth' are driven by one's desire to formulate a 'purpose' that reverts to the growth of experience with the intelligence of one's mind. In this context, the adults

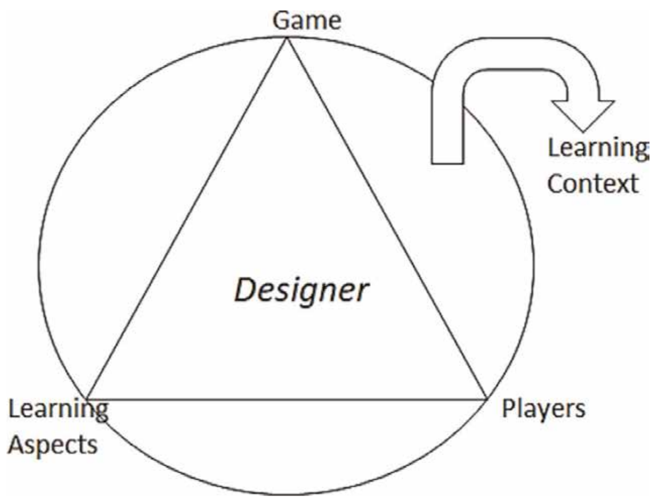


Figure 1.
The Designers' Magic Triangle (left). (Adopted from [12].)

whose learning methods combine work and studying, or theoretical and practical exposures are highly familiar and productive in their expertise.

The *purpose* is simple when a desirable quality " … which identifies freedom with power to frame purposes and to execute or carry into effect purposes so framed. Such *freedom* is, in turn, identical with *self-control*; for the *formation of purposes* and the organization of means them are the work of intelligence" ([50], p. 67). Dewey's description of an actual purpose is that it begins with an impulse. It is always equipped with "end-view"—which is involving "*the foresight of the consequences that may result from acting upon that impulse*" ([50], p. 67). Game designers create meaning and constantly reflect through their actions as the games are play-tested, designed, prototyped, iterated, or conceptualized. Hence, deriving an enriched experience shows an individualized aspect of comprehending how game designers differentiate games and EGs, how they distinguish between those above, and their dilemmas in the decision-making process that necessarily require their attention.

The principle of *Continuity* is where each experience taken from the past affects the future experience of the individual. In contrast, interaction refers to the situational and game designers' influence on players' or students' experiences. Dewey believes that the elements of *interaction* and the situation occur concurrently. Some challenges may be encountered despite the importance of experience, such as "that all experiences are genuinely or equally educative" ([50], p. 13). Hence, the game designers must first comprehend the nature of human experience, stating that the effect judges the experience's value that this experience has on the individual's present and future, an individual's ability to contribute to society. Therefore, it is directly relevant to how game designers evaluate players' experience while playing the game. In this context, it does not matter whether the player is a gamer or a learner. This is placed directly to the idea of how game designers interact with every individual's needs, to be able to support their cultural and social differences, and to allow them to learn/play equally and freely. Experience, which may be enjoyable, may not contribute to personal growth or experience with no coherence to a situation that does not necessarily result in a cumulative learning foundation for future learning.

The theory of experience has been discussed and applied in many types of fields, and the prominent Kolb's cyclic model [51] of experiential theory was built on the works of Dewey and has provided its means in the field of education and EGs. As described above, experiential learning is simply defined from the perspective of how learners experience a phenomenon and use that to facilitate learning. It has become a mutual ground for both teachers and game designers attempting to integrate game-based learning with education [10, 51, 52]:

- Learning involves participation in the real-world
- There are intimate relations between experience and education.
- Understandings are derived from and modified through experience.
- Meaningful learning consists of action and reflection.
- Experiential learning is also based on the belief that people learn best by doing.

3.1 Participant recruitment criteria

Initially, the location was set to Victoria state, Australia; however, due to the pandemic, location expansion was obligatory. With criterion and snowballing recruitment procedures, semi-structured qualitative interviews with 17 game designers in Australia were completed (January 2020–October 2020). The criteria set for the recruitment process are shown in **Table 1**. Each interview session is approximately 2 hours and 30 minutes (Zoom, Discord, Google meet, MS Teams). This paper interprets one central theme (from six themes) emerging through the coding analysis. Consequently, the phenomenological approach is adopted to allow the researcher to uncover the essence of lived experiences described as the qualitative researcher pursues to identify the phenomenon of human experience and then derive a description representing the experience's very nature [53]. The essence should not be viewed as a vague idea but rather the ability to identify the meaning embodied in the lived experience [54].

3.1.1 Data collection method and interpretation approach

The interviews were audio-recorded, transcribed, and time-stamped. An initial hand-coding approach was adopted before an in-depth thematic analysis using NVivo

Criteria	Participants' details
Location	Within Australia (Victoria, New South Wales, Queensland. South Australia)
Years of experience	Between 5 years—19 years in the gaming industry
Type of experience	Designing and developing games (all types of genres). Must have experience in designing EGs.
Types of player's demographics	All demographics. Primary school players' demographics are beneficial.

Table 1.
Participant recruitment criteria in this study.

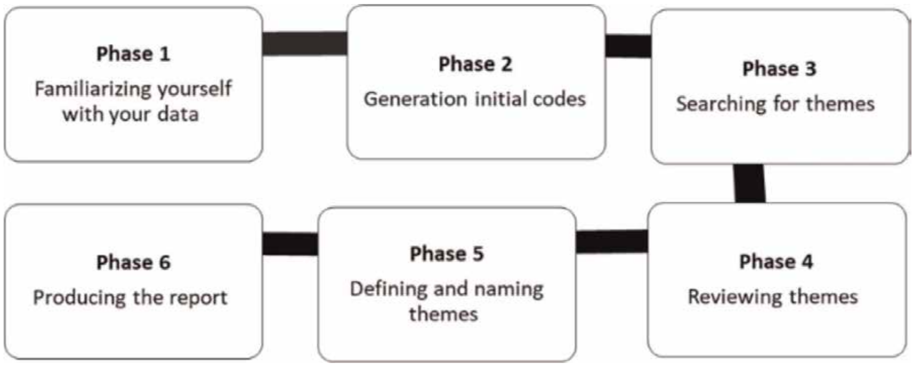


Figure 2.
Steps of thematic analysis. (Adopted from [55].)

10 software. This research adopted a six-step thematic analysis approach [55, 56], as shown in **Figure 2**. This analysis aims to allow game designers' perspectives discuss through reflective experience, their experience designing both types of gaming systems (its growth and continuity), and the way they criticize the conceptualization of games before they are designed (framing the purpose of designing games). Interpretive results are discussed through the interpretivism approach, where hermeneutic analysis is adopted to allow the subjectivity of game designers' experience. Consequently, the results are rigorously discussed to demonstrate experience in designing games and EGs, what they believe games and EGs have in common, and how they distinctively describe their professional opinions. Lastly, the analysis also correlates and is critically interpreted through researchers' epistemology and ontology position as well as concepts of the theory of experience.

4. Interpretive results discussion

This section discusses and interprets the central theme and sub-theme to demonstrate the perspectives of game designers' towards how they approach towards designing games in their daily routine as game designers. The following question: **how do they generally approach designing games and what are their go-to methods/approaches/considerations?** Understanding game designers' approaches and processes (throughout the lifecycle of the game), including any starting steps or crucial steps that they consider throughout their design processes are presented in this theme.

There are two sub-themes identified and categorized to narrow down the following discussion:

1. Formal approaches (game designers explicitly mentioned an approach/model), and

2. Personal experience-based designing approaches (their practice-based experiences, and intuitively adapting to the requirement of the game).

Please note that all names mentioned in the analysis are pseudonyms. The validity consideration of data analysis towards the authenticity and accuracy throughout the research is achieved through techniques including triangulation (examining with theoretical concepts, investigator, and data triangulation), and peer debriefing, prolonged and persistent engagement with the raw data set to allow critical and clarity throughout the analysis and emergence of themes to be compared.

4.1 Formal approaches and processes of game designing

Seven game designer participants mentioned a specific method when asked how they approach the game design process in relation to a specific model or method:

1. Math concept,
2. Game jamming,
3. Three broad ways (via a publisher, solo indie development, and client's feedback on ideas),
4. Top-down (traditional software design),
5. Bottom-up (non-traditional software design but still effective),
6. Conceptualization from pre-production,
7. Three spaces (theme, mechanics, and props).

With seven uniquely identified game design approaches in the data analysis, it is crucial to describe their unique position, techniques, or steps individually. These methods were distinctively approached with constraint (*social control*) that influences their decision-making. These constraints are from within their studios or their central pillars/rules or in the timeline in relation to their budget, and the complete design processes must be reached despite these constraints. Hence, I observed that game designers were reflecting towards the situation where they work, they are obliged to adapt into that social and collaborative environment (*freedom*), where their ideas develop throughout the process (*continuity*) to ensure that they complete their collaborations through teamwork and that their belief in exchanging ideas and concepts was thoroughly demonstrated through their explanations; The *growth* of experience based on a purpose.

I begin with the **math concept** approach and how it assists game designers. It is known as a fundamental foundation of successful game design, and it is commonly adopted in software engineering. It allows the ability to provide a calculative trajectory in correspondence to the player's choices and ensures that mechanics work entirely in response to the player's interactions with the game. As metaphorically suggested by video game developer "*It is the flour to the cake that game developers are trying to bake. Without it, the cake wouldn't rise*" [57]. In addition, Stenquist [57] demonstrated that the Math Concept allows the game design to become 'whole': animation, game engine architecture, gameplay scripting, analyzing player's interaction, timers, physics, graphics, path findings, etc., are all achieved through the Math Concept. For instance, Betty explained:

> "*Starting with an idea, or math concept or a theme, which will have a math concept method, or we find the artwork, but we need an innovative idea to go with it. So, we*

use the math concept to create the procedural design by leveling our design through prototyping and iterations constantly, to visualise it as a whole game idea".

Game jamming was mentioned by Joel. Game Jams are also conducted yearly around the globe as a platform to provide game designers a way to rapidly prototype game design ideas. It inspires growth across the gaming industry through the use of a given theme and constraints [35, 58, 59], that are represented in annual game design competitions [60], and recognized as the world's most significant game development events, with approximately 150% growth per year [61]. For instance, Joel explained how this process allows game designers to develop their abilities to learn, challenge themselves, become creative, and work through macro and micro levels of design within time constraints. It can be observed that the continuous repetition of the initial purpose of the game within the situational constraints provided to game designers also demonstrates social control and freedom. Hence, their personal desires and inspirations to become a game designers allow their persistent willingness to take challenges and grow their experience as game designers believe that you learn by doing. As Dewey acknowledges that art is not disconnected from our everyday lives and experiences, it is an 'intensified' form of experience. Art as experience ([43], p. 3) seems to be something that happens; it is the experience of the artist doing the work and the audience receiving it [46]. As Joel indicates:

"*Usually, game jams or a quick mini game project. Game jamming process is associated with micro development challenges, small teams, and time allocation. Our interpretation of the theme expands to the game idea, we'll re-evaluate and try, iterative, test-driven with often emergent acumen. What part is fun, and then pivot to do that … brainstorming, and then in a real-world (a studio)*"

The **three broad ways** consist of (1) publisher, (2) solo indie development, and (3) client's feedback on ideas. For instance, Terry states: "Why are we making this game at all? What's the point? What are the circumstances in which the game is being made?" In addition, production constraints and the timeline of projects are influential in decision-making throughout the processes, Terry asserted: "*What is the context? Competitive games relevance? Target audience? De-constructing relevant games? And are there any core pillars to allow multiple decision-making to remain consistent*?" Understanding at a meta-macro level is necessary for game designers to understand the concept and explore its possibilities. The concept is developed and is then in need of an overall game design. For instance, Terry explained that there is usually a lack of players or target audience available for game ideas to be tested. Therefore, a website called Quantic Foundry provides detailed research into different demographics and preferences for different styles of gameplay [62]. This positively assists their game design process.

As mentioned by Terry: "it's about adding continuous details, layers of details … articulate it in words … images … iterate … prototype … " This process reconnects with how the continuity, interaction, and freedom concepts (Dewey) can be positioned in designing games. In addition, game designers' developing experience increases versatility in game ideas. The situation in which they comprehend and evaluate the impacts of their decisions on how they approach games design is complicated and time constrained. As shown, it is a 'rolling-the-ball' scenario type: to visualize what lead to the desired outcome and what hindered the process. As Terry quotes:

> *"… [Questions stated above] … do a deconstruction of the competition to know what the conventions are […] correlating with the defined design pillars or core pillars, so, selecting central tenets that you repeatedly encounter as you're making decisions. Identify its supporting elements and correlating with initial idea [discuss with team] … ideally, get feedback from people outside of the team to get fresh perspectives."*

The **top-down approach** is known as the traditional game design approach [63], where game designers indulge in defining and abstracting an idea which is the foundational concept for the game at its initial stage: it breaks down its game style, setting, and plot motivations within the concept of the game including story elements, fiction and esthetics, etc., ultimately demonstrating a cohesive game design (see Raymond's extract below). To begin with, it is focused on the concepts (vision) and context of a game; as the layers progress in the model, the game-related functions are clarified. As Raymond explained, there is a need to have a clear and detailed definition of crucial game features, including mechanics, rules, and interactive action elements to allow an accurate picture to emerge through the process. To provide verbs that manipulate the system and provide feedback to the player and the game world, as actions are performed. Hence, the desired concepts, goals, and features are built. This model is available in Gamasutra [63]. The layered view of a Game design model [63], is presented in **Figure 3**. As Raymond quotes:

> *"… A lot of it is primarily top-down approach, starting with a vision of the game. It's an admission that involves fictional worlds, story elements, some interactive action elements. It's what you might imagine being a typically inspired response to playing games … so all my play experiences of many games begin to envision things of the whole product."*

The **bottom-up approach** is further mentioned by Raymond and Charlie. As shown in **Figure 3**, Raymond also uses the bottom-up approach as it allows him to

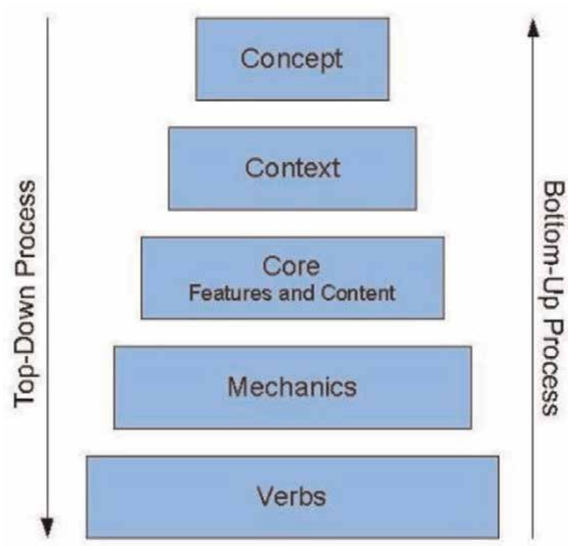


Figure 3.
Layered view of a game's design: top-down and bottom-up process. (Adopted from [63].)

begin with more minor interactive elements before the whole game system idea emerges: "*I have used both approaches and it depends on what the game really needs*". Charlie, on the other hand, she emphasized that: "*I feel it's very effective the bottom-up approach because we're starting with the minor interactions with small elements by defining and exploring what they are and that guides how the rest of the game's content and design holds out*". The top-down and bottom-up approaches are mainly found in software design, and these are commonly used terms [64], however, the interpretations vary in a different context, and the technical basis of the terms requires differences in starting points for designing software.

The sixth approach is called **conceptualisation** from pre-production (low-level to high-level design), which was highlighted by James. This approach is similar to how Adams [42] explained that there are three stages of the design process: (1) concept stage, (2) elaboration stage, and (3) tuning stage. This matches the game design approaches adopted by 10 of the game designer participants. It can be argued that these three stages are not applied by all 10 game designers in this study in the same fashion. However, the similarity must be addressed. I observed that James acknowledged this approach as: "*So, a specific reason to venture into designing/amending changes in game design. Sometimes, we encounter a specific design problem: how are we going to fix that?*". The concept stage usually requires a few weeks to sort the game ideas and test to ensure that the elaboration stage has begun. Here continuity and interaction through playtesting rectify the concept before the product develops a deeper sense of understanding and enables shared experiences through learning and engaging with other team members.

> *[James]: […] starting with a pre-production phase as designers. That's essentially like the 'incubation period.' […] start thinking about what it is that you want to do? Why? […] a reason to accomplish such as the need to improve this [existing game part], or we need to add features [design] … for further enhancement … Learning what players want more, brainstorming ideas [a high-level design to low-level design][3] … iterating details, reaffirming amongst designers, and then unfolding with the team to develop it.*

The last method suggested was **three spaces:** (1) theme, (2) mechanics (rules and interaction) and (3) prop. This method was described by Timothy that you begin designing a game on either of the three spaces and explore further the depths of core interactions, players' position in the design and rules, etc.; as explained in [65]. Here, concepts of interaction and situations are prominently observed. The game designers are inspired and create their desired 'intelligence' to form around the concept. The sole purpose is formed in parallel to the game system. With intelligence, the freedom that game designers must execute the purpose of designing games is practically framed throughout the procedure in parallel, resulting in a complex game design. As Timothy explains:

> *"[…] remembering older games' concepts and constantly reflecting on them to recreate them differently … that games are designed from one of three spaces: theme (like the fiction, game narrative), a mechanic (interaction of rules or rules like objects that you*

[3] Larger concepts of the game into more minor aspects of games—Generic terms used in software designing process.

wish to combine) or you have a prop (a material or toy that you use as inspiration for the game)!"

4.2 Personal experience-based designing approaches

According to the data, most game designers do not start a game from scratch. Joel mentioned: "It is highly unlikely that we begin from scratch!". They mostly begin with an idea, a concept, or a theme that they have been thinking or reflecting on. Sometimes, they reflect upon the best and the worst games played in their experience and took that idea forward [66]. As observed in the earlier sub-theme, other game designer participants mentioned either a theme, a concept, a mechanic, or a physical prop (tangible material) to start a game design approach.

This theme generalizes how 10 out of 17 participants had a similar approach to designing games. Their description to 'how they design games' was informal, inconsistent, interchangeable, and unpredictable; however, their overarching terms used to describe their processes included:

1. Conceptualization,

2. Reflective-based design experiences, and

3. Generalization of designing processes that were influenced by various formal methods. As they suggested that their beginning steps are to consider:

[Nora]: "self-learning or positioning themselves in the context",

[Blaire]: "conceptualisation and brainstorming",

[Ross]: "who is my audience?",

[Paige]: "Is there a theme or goal with rules?",

[Max]: "Brainstorming and experience expectations."

[Anne]: "Specific concept challenge and link mechanics and rule",

[Charlie]: "Starting with a verb."

[Tia]: "researching what exist and innovate",

[Zack]: "Ideation and mechanics",

[Warren]: "Concept art, conversations, themes, mental models".

Some published work supports certain processes such as **brainstorming** and **conceptualizing** approaches, along with techniques including ideation, prototyping, testing, player's psychology involved in designing games [18, 27, 42].

All game designers mentioned the process of conceptualisation in close relation to **prototyping** and **iteration**. Dewey positions art as experience as a part of the overall human experience [46]; that it *is born in the lives of the artists based on the conditions and societies that surround them*. This demonstrates the importance of art in experience and its relevance to the growth of experience [46, 55]. To quote Dewey ([43], p. 3):

> *"When an art product once attains classic status, it somehow becomes isolated from the human conditions under which it was brought into being and from the human consequences it engenders in actual life-experience."*

This section concludes with different approaches to game designing that the game design participants use in their practice.

5. Proposing Game Designers' Initiation tool

As the interpretive results discussion suggests that there are two broad approaches that game designers use for game designing: Formal methods and experience-based approach. To conceptualize these approaches, methods, and techniques that are suggested by the game designer participants, the representation of these methods and approaches are presented in **Figure 4**. This framework is called the *Game Designers' initiation tool*. As Dewey suggested in terms of generalization and registering the experiences individuals' have accounts for a more revolutionized and conceptualized understanding of a complex phenomenon such as the game design process. He states:

> *"The idea that generalization, purposes, etc., are individual mental processes did not originate until experience had registered such a change that the functions of the*

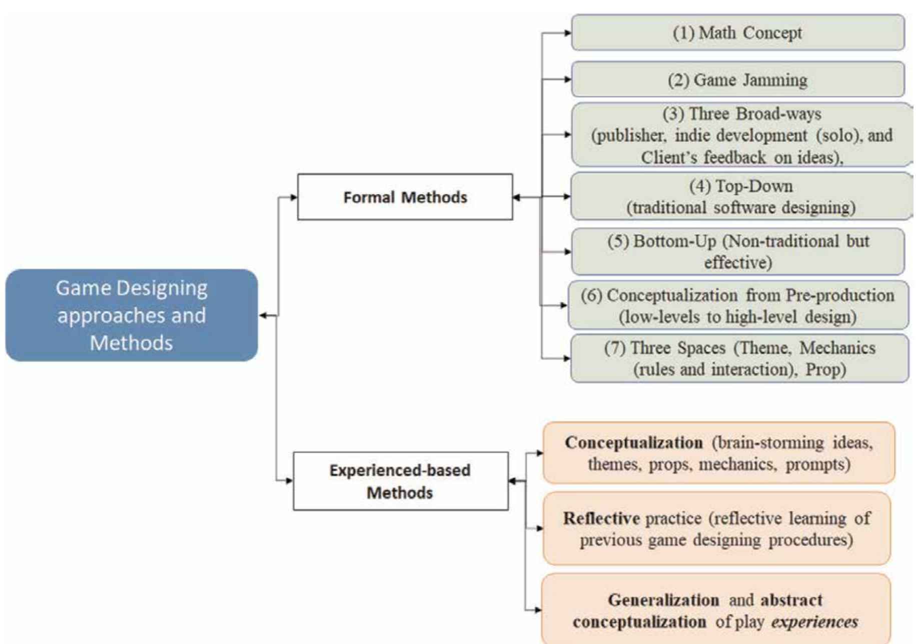


Figure 4.
Proposing Game Designers' initiation tool.

individualized mind were productive of objective achievements and hence, capable of external observation." ([66], p. 215)

6. Further recommendation

This chapter reports on a part of a larger study, however, it is crucial to identify and examine the difference in the approaches to designing games that are addressed by the game designers based in Australia and how that seems to be a challenging aspect in researching fields of game design and development. There is a need to ensure the inclusivity of game designers/developers from the industry into academia to help nurture the desired processes of game designing with the researching drive in the field. Moreover, I would urge the researchers in the field to explore a few more sub-areas through this lens of experiential learning:

1. Understanding the gaps between practices with researching game models and frameworks and whether those are useful for the game designers/developers.

2. What is actively adopted in the industry by the stakeholders and what is being proposed by the academia in the gaming field.

3. Finding a bridge between the two gaps mentioned to enable a better understanding across all game genre/fields including EGs, serious game, instructional game designs as well as other areas.

4. Finding effective data collection methods that can enable a deeper learning curve for researchers as they may potentially gain knowledge from experts in the field.

7. Conclusion

This chapter identifies the prominent game designing approaches and methods that the game designers in Australian gaming industry apply. Two prominent categories have emerged through the data analysis: Formal methods and experience-based methods. Game Designers' Initiation tool is proposed to assist game designers in their practice to begin their designing process. This is to demonstrate an easy projection on selecting the game design process based on what other factors are compiled into their game ideas. This research will continue to bridge the gap between various gaming genres.

Author details

Mifrah Ahmad
Deakin University, Australia

*Address all correspondence to: mifrah09@gmail.com

References

[1] Brand JE, Todhunter S, Jervis J. Digital Australia Report 2018 (DA18). Eveleigh, NSW: IGEA; 2017 Contract No.: 1-43

[2] Williams DR. Australia's Games Development Industry Is Now Worth $185 Million Kotaku Australia: 2022 Pedestrian Group; 2021. Available from: https://www.kotaku.com.au/2021/02/australia-igea-survey-jan-2021/amp/

[3] Brand JE, Jervis J, Huggins PM, Wilson TW. Digital Australia 2020: The Power of Games. Eveleigh, NSW: IGEA; 2019

[4] IGEA. Australian Video Game Development: An Industry Snapshot FY 2019-20: IGEA: Interactive Games & Entertainment Association; 2020. Available from: https://igea.net/wp-content/uploads/2021/01/AGD-2020-Infographic-2pp-SQUARE-1.pdf

[5] IGEA. Game Engine: Australia's immediate oppurtunity to create a $1 billion video game development industry Eveleigh, NSW 2015: IGEA: Interactive Games & Entertainment Association; 2020. Available from: https://igea.net/wp-content/uploads/2020/06/IGEA-Game-Engine-paper.pdf

[6] Curry R. Australian Game Development: An Industry Snapshot 2020-2021. 2021. Available from: https://igea.net/wp-content/uploads/2021/12/IGEA-AGD-Survey-Report-2021-22-Final.pdf

[7] Costikyan G, editor. I have no words & i must design: Toward a critical vocabulary for games. In: Proceedings of Computer Games and Digital Cultures Conference; 2005; Tampere: Tampere University Press; 2002

[8] Erhel S, Jamet E. Digital game-based learning: Impact of instructions and feedback on motivation and learning effectiveness. Computers & Education. 2013;**67**:156-167

[9] Stenros J. The game definition game: a review. Games and Culture. 2017;**12**: 499-520

[10] Salen K, Zimmerman E. Rules of Play: Game Design Fundamentals. Cambridge, Massachusetts: MIT Press; 2004

[11] Schell J. The Art of Game Design: A Book of Lenses. Burlington, USA: Morgan Kaufman; 2008. pp. 1-489

[12] Kalmpourtzis G. Educational Game Design Fundamentals: A Journey to Creating Intrinsically Motivating Learning Experiences. New York: AK Peters/CRC Press; 2018

[13] Keogh B. From aggressively formalised to intensely in/formalised: Accounting for a wider range of videogame development practices. Creative Industries Journal. 2019;**12**(1): 14-33

[14] Keogh B. The cultural field of video game production in Australia. Games and Culture. 2021;**16**(1):116-135

[15] Stieler-Hunt C. Advancing the Use of Digital Game-Play in Primary and Secondary School Classrooms to Establish Supportive and Engaging Classroom Learning Environments: University of the Sunshine Coast [Dissertation]. 2016. DOI: 10.25907/00265

[16] Kultima A, Lassheikki C, Park S, Kauppinen T, editors. Designing games as playable concepts: Five design values for tiny embedded educational games.

In: Proceedings of the 2020 DiGRA International Conference: Play Everywhere, Tampere, Finland. 2020

[17] Prestridge S, de Aldama C. A classification framework for exploring technology-enabled practice–FrameTEP. Journal of Educational Computing Research. 2016;**54**(7):901-921

[18] Crawford C. The Art of Computer Game Design. Osborne/McGraw-Hill; 1984. Available from: http://www.stonetronix.com/gamedesign/art_of_computer_game_design.pdf

[19] Zimmerman E. Play as research: The iterative design process. Design Research: Methods and Perspectives. 2003;**2003**:176-184

[20] Kultima A, editor Developers' perspectives on iteration in game development. Proceedings of the 19th International academic Mindtrek conference; 2015

[21] Cross N. From a design science to a design discipline: Understanding designerly ways of knowing and thinking. In: Michel R, editor. Design Research Now. Board of International Research in Design. Basel: Birkhäuser, Springer Link; 2007. pp. 41-54

[22] Zyda M. From visual simulation to virtual reality to games. Computer. 2005; **38**(9):25-32

[23] Harteveld C, Guimarães R, Mayer IS, Bidarra R. Balancing play, meaning and reality: The design philosophy of LEVEE PATROLLER. Simulation & Gaming. 2010;**41**(3):316-340

[24] Laine TH, Lindberg RS. Designing engaging games for education: A systematic literature review on game motivators and design principles. IEEE Transactions on Learning Technologies. 2020;**13**(4):804-821

[25] Fullerton T. Game Design Workshop: a Playcentric Approach to Creating Innovative. Games: CRC Press; 2014

[26] Perry D, De Maria R. Game design: A brainstorming toolbox. In: Hiquet SL, editor. Course Technology, a Part of Cengage Learning. 1st ed. Charles River ISE; 2009

[27] Fullerton T, editor. A play centric approach to creating innovative games. In: Morgan Kaufmann Game Design Books. CRC Press; 2008

[28] Kultima A. Game Design Praxiology. Acta Electronica Universitatis Tamperensis 1885. Tampere University Press; 2018. Available from: https://trepo.tuni.fi/bitstream/handle/10024/103315/978-952-03-0742-4.pdf

[29] Dingsøyr T, Dybå T, Moe NB. Agile software development: An introduction and overview. In: Agile Software Development: Current Research and Future Directions. Berlin, Heidelberg: Springer; 2010. pp. 1-13

[30] Keith C. Agile game development with Scrum. Pearson Education; 2010

[31] Adams RS, Atman C, editors. Characterizing engineering student design processes an illustration of iteration. In: 2000 Annual Conference. 2000

[32] Larman C, Basili VR. Iterative and incremental developments: A brief history. Cover Feature IEEE Computer. 2003;**36**(6):47-56

[33] Kultima A, editor. The organic nature of game ideation: Game ideas arise from solitude and mature by

bouncing. In: Proceedings of the International Academic Conference on the Future of Game Design and Technology. 2010

[34] Boehm BW. A spiral model of software development and enhancement. Computer. 1988;**21**(5): 61-72

[35] Kultima A, editor. An autopsy of the Global Game Jam 2012 theme committee discussion: Deciding on ouroboros. In: Proceedings of the 10th International Conference on the Foundations of Digital Games (FDG 2015). 2015

[36] Williams L. Agile software development methodologies and practices. In: Zelkowitz MV, editor. Advances in Computers. Vol. 80. Elsevier; 2010. pp. 1-44

[37] Kultima A, editor. Casual game design values. In: Proceedings of the 13th International MindTrek Conference: Everyday Life in the Ubiquitous Era. 2009

[38] Kultima A, Alha K. Hopefully everything I'm doing has to do with innovation: Games industry professionals on innovation in 2009. In: 2010 2nd International IEEE Consumer Electronics Society's Games Innovations Conference, Hong Kong, China. 2010. pp. 1-8. DOI: 10.1109/ICEGIC.2010. 5716899

[39] Hagen U. Designing for player experience: How professional game developers communicate design visions. Journal of Gaming & Virtual Worlds. 2011;**3**(3):259-275

[40] Swartout W, van Lent M. Making a game of system design. Communications of the ACM. 2003;**46**(7):32-39

[41] Hagen U. Where do game design ideas come from? Invention and recycling in games developed in Sweden. In: Breaking New Ground: Innovation in Games, Play, Practice and Theory. Proceedings of DIGRA. No. 24. 2009. pp. 1-11

[42] Adams E. Fundamentals of Game Design. 2nd ed. Pearson Education, Inc., New Riders Publishing; 2010

[43] Dewey J. Art as Experience. New York: First Perigee Printing; 1934

[44] Kultima A, Stenros J, editors. Designing games for everyone: the expanded game experience model. In: Proceedings of the International Academic Conference on the Future of Game Design and Technology. 2010

[45] Zagal JP, Nussbaum M, Rosas R. A model to support the design of multiplayer games. Presence: Teleoperators & Virtual Environments. 2000;**9**(5):448-462

[46] Wilson D, Sicart M, editors. Now it's personal: on abusive game design. In: Proceedings of the International Academic Conference on the Future of Game Design and Technology. 2010

[47] Blessing LT, Chakrabarti A. DRM: A Design Reseach Methodology. London: Springer; 2009. pp. 13-42. DOI: 10.1007/ 978-1-84882-587-1_2

[48] Björk S, editor. Games, gamers, and gaming: Understanding game research. In: Proceedings of the 12th International Conference on Entertainment and Media in the Ubiquitous Era. 2008

[49] Bjork S, Holopainen J. Patterns in Game Design (Game Development Series). Charles River Media, ACM Digital Library, Inc.; 2004

[50] Dewey J. Experience and education. In: Hall-Quest AL, editor. Kappa Delta

Pi. First Touchstrone Edition 1997 ed. Rochefeller Center, New York, NY 10020: Touchstone NY; 1938. pp. 1-99

[51] Kolb DA. Experiential Learning: Experience as the Source of Learning and Development. Englewood Cliffs, NJ: Prentice Hall; 1984

[52] Appelman R. Designing experiential modes: A key focus for immersive learning environments. TechTrends. 2005;**49**(3):64-74

[53] Kafle NP. Hermeneutic phenomenological research method simplified. 2011:181-200

[54] Merleau-Ponty M, Smith C. Phenomenology of Perception. London: Routledge; 1962

[55] Braun V, Clarke V. Using thematic analysis in psychology. Qualitative Research in Psychology. 2006;**3**(2):77-101

[56] Braun V, Clarke V. Thematic analysis. In: Cooper H, Camic PM, Long DL, Panter AT, Rindskopf D, Sher KJ, editors. APA Handbook of Research Methods in Psychology, Vol. 2. Research Designs: Quantitative, Qualitative, Neuropsychological, and Biological. American Psychological Association; 2012. pp. 57-71. DOI: 10.1037/13620-004

[57] Stenquist MS. This Is The Math Behind Super Mario Quora Contributor: Forbes. 21 October 2016. Available from: https://www.forbes.com/sites/quora/2016/10/21/this-is-the-math-behind-super-mario/?sh=1518a8c42154

[58] Jam GG. GJ 2014 by the Numbers 2014. Available from: http://globalgamejam.org/news/ggj-2014-numbers

[59] Meriläinen M, Aurava R, Kultima A, Stenros J. Game jams for learning and teaching: A review. International Journal of Game-Based Learning (IJGBL). 2020;**10**(2):54-71

[60] Ho X, Tomitsch M, Bednarz T. Game design inspiration in global game jam. In: Proceedings of DiGRAA. 2014

[61] Fowler A, Khosmood F, Arya A, editors. The evolution and significance of the Global Game Jam. In: Proc of the Foundations of Digital Games Conference. 2013

[62] Foundry Q. The Science of Gamer Motivation: Quantic Foundry. 2015-2021. Available from: https://quanticfoundry.com/

[63] Kuhnen R. Game Design Cognition: The Bottom-Up and Top-Down Approaches. 14 November 2007. Available from: https://www.gamedeveloper.com/design/game-design-cognition-the-bottom-up-and-top-down-approaches

[64] Pizka M, Bauer A, editors. A brief top-down and bottom-up philosophy on software evolution. In: Proceedings 7th International Workshop on Principles of Software Evolution, 2004; IEEE. 2004

[65] Hunicke R, LeBlanc M, Zubek R. MDA: A Formal Approach to Game Design and Game Research. Workshop on Challenges in Game AI. 2004. pp. 1-4

[66] Dewey J. In: Carus P, editor. Experience and Nature. London: George Allen & Unwin Ltd: Courier Corporation; 1929

Chapter 4

Automatic Generation of Game Content Customized by Players: Generate 3D Game Characters Based on Pictures

Hsuan-min Wang, Cheng-Lun Tsai and Chuen-Tsai Sun

Abstract

With the advancements in hardware, deep learning, and the invention of generative adversarial networks (GANs), the integration of games and artificial intelligence (AI) has mostly focused on assisting game development, such as creating maps, skills, monsters, NPCs, and levels. In this chapter, a different approach is proposed, which utilizes artificial intelligence in games to allow players to customize game content. Based on this concept, a method for automatically generating 3D game character models using images is presented, specifically the Parallel PIFu (Pixel-Aligned Implicit Function) model. This method leverages the characteristics of the PIFu model and combines the features of the generated 3D models from the front and back views of the person captured in the images. By merging these features, the method produces 3D character models that preserve the details from both images without any missing body parts. This approach builds upon existing techniques for automatic generation of character models and further enhances them. It enables users to simply use their mobile phones to capture images of their desired characters, which can then be automatically transformed into corresponding 3D models. These models are compatible with most existing games on the market, allowing players to easily create personalized appearances for their in-game characters and enhance the overall gaming experience.

Keywords: AI in games, player customization, 3D game character models, immersive gaming experience, deep learning

1. Introduction

With the evolution of technology and advancements in hardware devices, the gaming industry has witnessed continuous innovation. From classic games like Snake and Space Invaders to modern titles such as Monster Hunter and League of Legends, the gameplay and content of games have expanded significantly. This is largely attributed to game designers who incorporate imaginative ideas to create unique and enjoyable gaming experiences for players. As artificial intelligence (AI) technology matures, game developers have started leveraging AI not only as non-player characters (NPCs) but also across various elements of the game, aiming to enhance player enjoyment and enrich the game's content.

In order to enhance gameplay and player enjoyment, many game developers have shifted their focus from 2D games to 3D games. However, this also brings a challenge for game developers to find an effective way to generate a large number of 3D characters.

When converting 2D models to 3D models, developers need to address how to preserve the proportionate details of the characters and maintain realism and physicality during character movements. They also need to ensure consistency in the model from multiple viewing angles. Consequently, the process of 2D to 3D conversion involves a significant amount of modeling work. This not only consumes a considerable amount of time but also incurs corresponding costs in terms of hardware resources. Despite the existence of AI-generated techniques, most current research focuses on facial model modifications, and there is relatively less functionality available for full-body modeling. As a result, the templates provided for automatic adjustments in games mostly concentrate on facial features.

Currently, many games incorporate a step where players can create their own virtual characters before starting the beginner's tutorial. This seemingly simple step is a crucial aspect of allowing players to bring their own experiences into the game. Recently, many games have introduced systems that allow players to freely adjust parameters, enriching the appearance possibilities in the game. However, at times, having too many adjustable parameters can lead to the problem of spending excessive time on character creation.

Therefore, this chapter proposes the concept of combining AI-generated content with player customization in games and, based on this concept, introduces a method for generating 3D game character models from images. The results demonstrate that this method can effectively generate 3D character models corresponding to casually captured photos, which are compatible with the majority of games on the market. It allows players to easily create their own character appearances in the game and provides game developers with a tool for 3D character modeling.

2. Literature review

In order to generate 3D characters that can be applied to games from images, this section involves body shape and posture prediction, 3D model generation, and human skeleton generation. The above-mentioned related technologies will be discussed below.

2.1 A Skinned Multi-Person Linear Model (SMPL)

The Skinned Multi-Person Linear Model (SMPL) [1] is a data-driven approach that uses a base human body model to simulate various body shapes and poses through extensive training. This model allows for deformations of the base model by utilizing low-dimensional parameters obtained through principal component analysis (PCA), which represents variations in height, weight, and other body attributes.

2.2 Human Mesh Recovery (HMR)

The Human Mesh Recovery (HMR) model [2] is a 2D-to-3D model that can reconstruct a 3D human model from input photos containing human subjects. The architecture of HMR integrates feature extraction, the SMPL model, and GAN models. HMR takes image features as inputs to the SMPL model, which then infers the 3D human model. The inferred model is projected back onto the 2D image to compute the loss. To ensure that the generated poses are realistic and satisfy the constraints of 3D joint angles, HMR employs a GAN discriminator to distinguish between the generated model and real human models. This helps improve the quality of the generated results, making them more consistent with real human movements. Additionally, during training, HMR does not require paired 2D images and their corresponding 3D models for supervised learning.

2.3 Generating 3D models of clothed characters from 2D images

In the realm of generating 3D models of clothed human bodies, notable advancements have been made, particularly with the Pixel-Aligned Implicit Function (PIFu) approach [3–5]. PIFu excels in inferring the surface details of single and multiple views of an object, showcasing outstanding performance.

There are two common representations for object surfaces: (1) explicit surfaces and (2) implicit surfaces. Explicit surfaces directly provide the points on the surface or use mapping relations to obtain them, while implicit surfaces do not explicitly represent the points but describe the relationship that all points on the surface satisfy. Each representation has its pros and cons. Explicit surfaces allow easy sampling of all points but require significant storage space and cannot determine the relationship between an arbitrary point and the surface. Implicit surfaces, on the other hand, can easily determine the relationship between any point and the surface, require less storage space, but are challenging to sample all points on the surface. In PIFu, an implicit function is utilized to define the object surface [6] due to its storage efficiency.

Although there have been previous attempts to generate 3D models by inferring object surfaces using implicit functions [7–9], purely relying on implicit functions cannot accurately capture the details in the input images. To address this, the concept of Pixel-Aligned is introduced. It involves incorporating the features extracted from the corresponding positions on the input images using a fully convolutional image encoder and a multi-layer perceptron (MLP) into the implicit function.

Compared to solely learning the implicit function in 3D space using global features, incorporating image features helps preserve the local details present in the images. However, PIFu has its limitations in practical applications. When reconstructing 3D models from a single view, although it avoids generating fragmented models, the reconstruction of the side profile of a person or the unobserved side not present in the photo is not ideal. When reconstructing from multiple views, while the overall reconstruction is more complete compared to a single view, it relies on using different-angle photos that have the same camera perspective and distance to the target object. Even slight perturbations can lead to less ideal 3D reconstructions, potentially missing important body parts such as hands and feet.

2.4 Automatic generation of 3D character skeletons

Automatic skeleton prediction can be divided into two main methods: (1) skeleton embedding and (2) skeleton extraction. In the skeleton embedding approach, the primary goal is to construct the geometric structure of the input model and define a penalty function for skeleton embedding. The predefined skeleton template is then embedded into the geometric structure to optimize the skeleton and obtain the final result. Pinocchio [10] is the most representative method using this approach. On the other hand, skeleton extraction is a more commonly used method for automatic skeleton prediction. It involves inferring the skeleton that fits the model using specific algorithms [11–14] or deep learning models [15, 16].

Both methods have their own advantages and disadvantages. The advantage of skeleton embedding is that it fully utilizes prior knowledge of the skeleton and does not rely solely on geometric topology. For example, the ear region does not require skeleton binding. However, due to the distinct geometric features, the skeleton extraction method may generate skeleton structures that do not correspond to the intended movements. A notable drawback of skeleton embedding is the need for prior knowledge of the skeleton prototype corresponding to the 3D model actions. On the other hand, the advantage of skeleton extraction is that it can handle a wide range of 3D models as inputs, such as birds, dinosaurs, and humans. Consequently, most related research focuses on skeleton extraction. However, the downside is that it may generate skeleton structures that do not align with the intended movements due to prominent geometric features. Since this chapter focuses on human models, the skeleton embedding method will be adopted.

In this chapter, the integration and improvement of the aforementioned techniques are employed to generate 3D character models suitable for use in games. Firstly, the HMR and PIFu models are utilized to generate 3D human models with clothing. The HMR model estimates the 3D pose and shape of the human body from a single 2D image, while the PIFu model performs detailed 3D reconstruction of the human body, including clothing, based on a single 2D image. This enables the generation of high-quality 3D human models with clothing.

Next, the generated 3D models undergo skeleton embedding. The skeleton embedding method, mentioned earlier, is employed to embed a predefined skeleton template into the geometric structure of the 3D models. Optimization using penalty functions is applied to achieve the final skeleton structure.

Finally, the skeleton is bound to the 3D models, and animations are created using the built-in environment of Maya. Maya is widely used software for 3D modeling and animation, offering a rich set of tools and features for skeleton binding, keyframe animation setup, and smooth animation effects. By performing skeleton binding and animation setup in Maya, interactive 3D character models with realistic movements can be created. Ultimately, the generated 3D human models with animations are exported for direct usage in games, enhancing realism and interactivity within the gaming environment.

3. Methodology

The methodology employed in this chapter is a combined model based on three components: Human Mesh Recovery (HMR), Pixel-Aligned Implicit Function (PIFu), and Pinocchio. By using this integrated model, users can generate 3D animated models of individuals by providing front and back photos. This approach facilitates the easy and automated generation of customized game characters (**Figure 1**).

3.1 Generation of 3D character models

The process of generating 3D character models can be divided into three stages: parallel PIFu modeling, HMR modeling, and refinement of PIFu results using HMR.

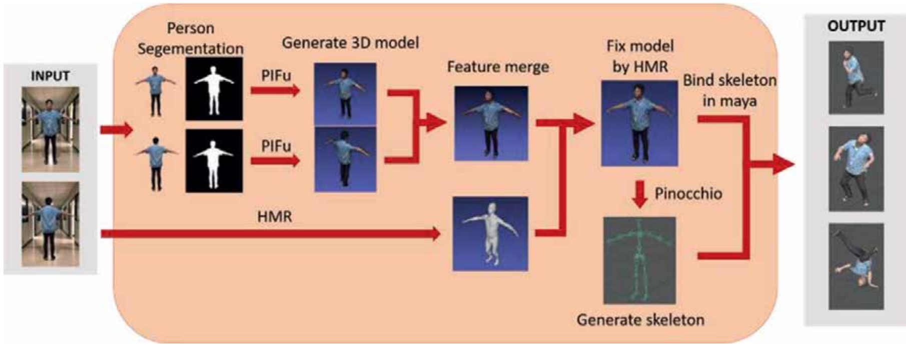


Figure 1.
The overall architecture of the model used in this chapter.

3.1.1 Parallel PIFu model

To create 3D models while preserving the detailed information of the characters in the images, a parallel PIFu model is used in this section. The architecture is based on the single-view PIFu approach, which reconstructs 3D models from a single image. However, using a single image alone cannot capture the complete model. Therefore, a new approach called parallel single-view is proposed in this chapter. This method combines the features of two 3D models generated from the front and back images of a person, maintaining the details from both images and ensuring a complete representation of the body.

The RenderPeople dataset [17] is employed in this method, which provides 491 high-resolution clothed human body models. The training and testing sets consist of 442 and 49 models, respectively. In terms of data processing, since the training requires input images along with their corresponding 3D models, RenderPeople only provides the models without corresponding images. To address this, the models are placed at the center and rotated 360 degrees to render them into images. In other words, a total of 442 × 360 = 159,120 images are generated as training data. However, since the goal is to reconstruct real-world scene images, the rendered images need to consider the effects of lighting. To simulate lighting accurately, a radiance transfer technique (PRT) [18] is utilized to precompute the global illumination effects, resulting in rendered images that closely resemble real-world scenes.

After the initial training of the PIFu model is completed, the parallel PIFu model training is conducted. The features from the front and back models are combined after the PIFu output. Since the output format of this model is in OBJ format, there is no existing method for feature combination. Therefore, a simple method is designed in this section to handle feature combinations. First, both models are uniformly scaled and aligned to the center. Then, the models are segmented into multiple parts, and the features from the front and back are merged by finding corresponding points. This approach enables the generation of a 3D model that includes features from both front and back images.

3.1.2 HMR model

In this stage, the training data for the HMR model consists of 2D and 3D data. The 2D training data includes datasets such as LSP, LSP-extended [19], MPII [20], and MS

COCO [21]. The purpose of using these datasets is to train the model to extract image features and generate 3D models that can be projected onto 2D models to compensate for the computational losses. The 3D training data includes Human3.6 M [22] and MPI-INF-3DHP [23], which are used to train the discriminator to distinguish between generated 3D models and real human models, and to generate corresponding 3D models.

The HMR model architecture consists of three parts: encoder, 3D regression, and discriminator. The encoder utilizes the ResNet-50 neural network architecture to extract features from the input images. The 3D regression part is a fully connected network that combines the image features and initial SMPL (Skinned Multi-Person Linear) body parameters. It generates body parameters that match the pose of the person in the image through iterative error feedback and calculates the loss by projecting the 3D joint nodes onto the image. The discriminator consists of shape discriminator and pose discriminator, which are used to determine whether the generated 3D model is real or fake. Finally, the adversarial loss is used to refine the parameters of the encoder.

3.1.3 Model refinement

Since the HMR model continuously refines the rotational deformation from the average body model parameters of SMPL, the generated model from HMR not only accurately represents the motion of the person in the image but also avoids the depth prediction errors that cause hand distortion as observed in PIFu. Leveraging this advantage, this method scales and aligns both models proportionally and fine-tunes the HMR model based on the average position of the hands in PIFu. Finally, the coordinate positions of the hand vertices in the HMR model are combined with the features from the parallel PIFu model to perform feature merging. This approach corrects the issue of hand distortion while preserving the original features of the vertices.

3.2 Skeleton generation

The 3D models generated in this chapter are all humanoid. Given the prior knowledge that the skeleton prototype is humanoid, it is ideal to use skeleton embedding to predict the skeleton positions. Therefore, this method employs the Pinocchio algorithm to predict the positions of the human body skeleton and output the coordinates of the joints and their corresponding relationships.

The Pinocchio algorithm can be divided into five main steps: (1) constructing the geometric structure graph that fits the model, (2) simplifying the input skeleton graph, (3) training the weights of different loss functions, (4) obtaining the optimal skeleton embedding result, and (5) refining the optimal result.

First, to obtain the geometric structure graph of the 3D model, the model is scaled to fit within a unit cube. Then, an octree structure is recursively applied to partition the model's surface until it can fully describe the geometry. With the surface information, the medial surface is estimated by sampling points discretely, and then, the interior space of the model is filled by maximizing spheres centered at the points on the medial surface. Finally, one of the following conditions is used as a criterion: (1) if two spheres intersect, an edge is constructed by connecting their centers, or (2) if two spheres do not intersect but their centerline is located inside the model and constitutes an important edge in the skeleton structure, the centers of the spheres are connected to form the geometric structure graph of the model.

The second step is to optimize the computation of the skeleton embedding. In this step, the input skeleton is simplified by merging all the joints with a degree of 2. This reduces the number of joints from the original 18 to 7. The simplified skeleton is then embedded into the geometric structure graph. After the embedding is completed, the shortest paths between the joints on the geometric structure graph are computed. The simplified skeleton structure is then restored based on the original input skeleton's proportions using interpolation.

The third step aims to achieve better skeleton embedding results. To do this, an appropriate loss function is defined for the skeleton embedding. Since a single loss function is not sufficient to cover all the constraints of the skeleton, nine different loss functions are defined [24]. These loss functions correspond to different constraints: (1) short bones: avoiding excessively short bones, (2) improper orientation between joints: ensuring consistency in the orientation of the input skeleton and the embedded skeleton, (3) length differences in bones marked symmetric: avoiding inconsistent bone lengths in symmetric pairs, (4) bone chains sharing vertices: avoiding bone chains that share vertices, (5) feet away from the bottom: ensuring that the joint nodes representing the feet are located at the bottom of the model, (6) zero-length bone chains: avoiding bone chains with zero length, (7) improper orientation of bones: ensuring consistency in the orientation of the restored simplified skeleton and the original skeleton, (8) degree-one joints not embedded at extreme vertices: ensuring that degree-one nodes are sufficiently far from their parent nodes, and (9) joints far along bone-chains but close in the graph: avoiding nodes that are far along the path but close in physical distance. The weights of these loss functions are trained using SVM, and the final loss function is obtained by linear combination.

Although the loss functions are defined, optimizing the loss function directly becomes challenging as the number of skeleton joints increases exponentially. Therefore, the branch and bound algorithm is used to estimate a lower bound for the skeleton embedding in advance, and then, the A-star algorithm is used to gradually add the embedded skeleton joints. Finally, the optimal skeleton embedding result with the minimal loss is obtained on the geometric structure graph.

After obtaining the optimal embedding result, the simplified skeleton with seven joints needs to be restored to 18 joints. The shortest paths between the joints in the geometric structure graph are found, and the restored skeleton is obtained by interpolating based on the proportions of the original input skeleton. Degree-2 joints are added to restore the same structure as the original embedded skeleton. Although the overall result is already good, there are still some local issues mainly due to the smaller weights assigned to smaller skeletons during the optimization process. Therefore, fine-tuning is required. A continuity optimization function is defined for this purpose. To facilitate fine-tuning, a continuous optimization function is defined as follows:

$$g(q_1,\cdots q_s)=\alpha_A g^A(q_1,\cdots q_s)+\sum_{i=2}^{s} g_i\left(q_i,q_{pS(i)}\right)$$

Where $(q_1,,\ldots,,q_s)$,s $=18$ represents the embedded skeleton's nodes, $P_{S(i)}$ denotes the parent node of the *i*th joint. The first term $\acute{a}_A g^A(q_1,,\ldots,,q_s)$ represents the penalty for asymmetry. The second term $g_i=\acute{a}_S g_i^S+\acute{a}_L g_i^L+\acute{a}_O g_i^O$ represents penalties for a skeleton being far from the central surface, having bones that are too short, and having a different direction from the original skeleton after embedding, respectively.

The final result of skeleton embedding is obtained by fine-tuning through the optimization function described above.

3.3 Skeleton binding and output

The final step is to bind and output the 3D model with the skeleton. In this method, Maya is used for skeleton binding and exporting in the FBX format. The chosen method for skeleton binding is Geodesic Voxel Bind, which is widely used. After binding the skeleton to the 3D model, the model can be animated according to the user's needs, based on a timeline sequence. The reason for using the FBX format for output is that it is a common model format used in various 3D modeling software and game engines available in the market.

4. Results

4.1 Results of the parallel PIFu model

In this stage, by incorporating feature fusion into the base PIFu model, we are able to address the limitation of single-view PIFu models, which only capture features from one side, as well as the model corruption issue that occurs in multi-view PIFu models when the photos have different camera perspectives and target object distances. The following are the results of this stage (**Figure 2**).

4.2 Model refinement results

In this stage, the HMR (Human Mesh Recovery) model was employed to iteratively refine the rotational deformation of the average human body model, resulting in a reconstructed human body model that aligns with the body shape and movements observed in the input photos. This refined model was used to correct the distorted deformations in the hands observed in the parallel PIFu model. Additionally, a feature fusion approach was applied to preserve the distinctive features of the hands. The following are the results achieved in this stage (**Figure 3**).

4.3 Overall model results

In this stage, the final model generation was performed by integrating the parallel PIFu model, the refined model, and the Pinocchio model into an automated generation framework. This framework enables users to easily create 3D characters for game development. The following presents the final results obtained using our proposed method (**Figure 4**).

4.4 Limitations

Although the method presented in this chapter is capable of generating 3D models of characters, there are several limitations that need to be acknowledged.

1. Restrictions on capturing poses: During the photo capturing process, the photographed person needs to maintain a T-pose with hands not touching the body and legs spread apart without merging. This is because the model interprets the

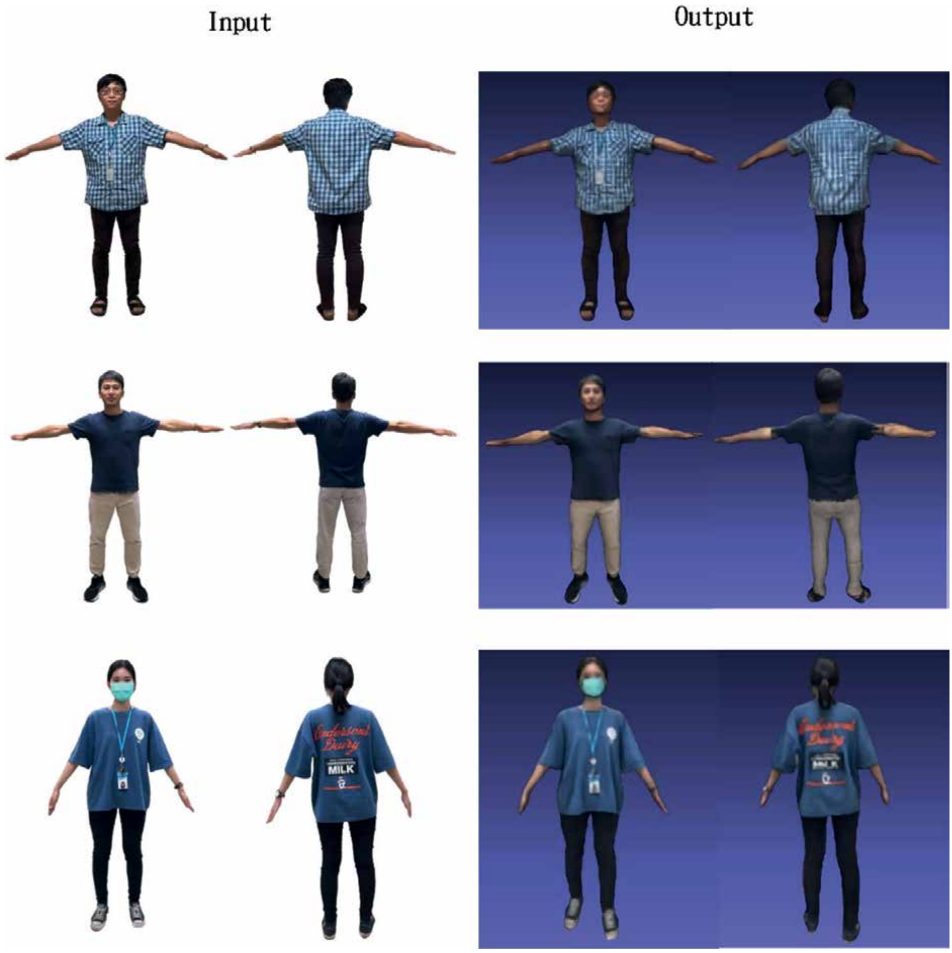


Figure 2.
Results of the parallel PIFu model.

connected regions between hands and body as a single entity, resulting in the generated character's limbs sticking to the clothing without separate movements.

2. Limitations in capturing fine details: While the overall reconstruction is satisfactory, limitations in hardware capabilities and limited training data prevent the use of high-resolution photos. As a result, the prediction of fine details may not be ideal.

3. Restrictions related to clothing: The corrective model used in the approach is based on unclothed human body models. Therefore, it cannot accurately correct the distorted parts of the hands of characters wearing long-sleeved clothing. This means that the proposed model is only suitable for characters wearing short-sleeved garments.

These limitations indicate areas for potential future improvements and research focus, such as addressing pose restrictions, enhancing the prediction of fine details, and extending the corrective model to handle different types of clothing.

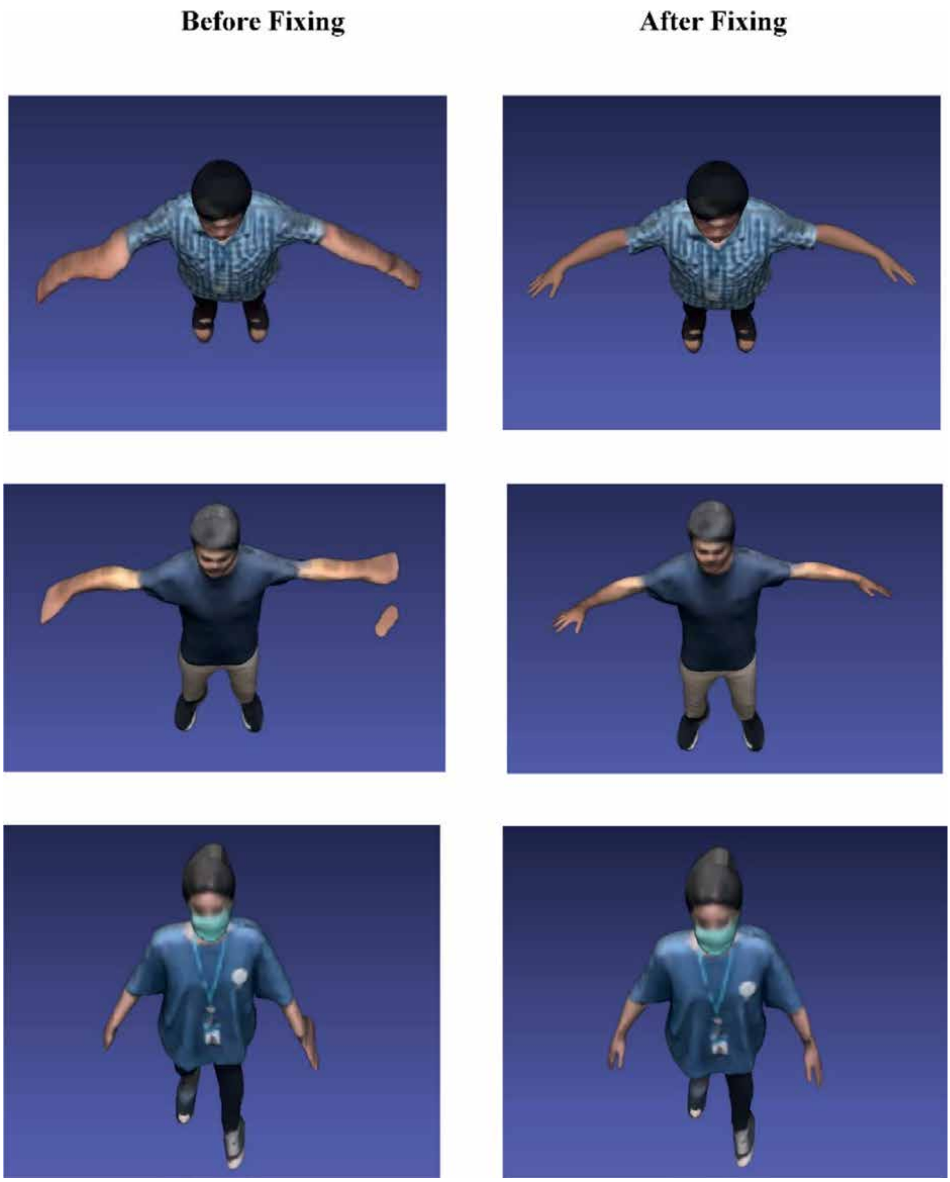


Figure 3.
Results of model refinement.

5. Conclusion

This chapter introduced a novel concept that combines the elements of automatic generation and customization in games, along with an automated model generation framework for customizing character appearances. With this framework, players can simply capture front and back photos of the desired character, and the system will automatically generate a character model that can be used in a

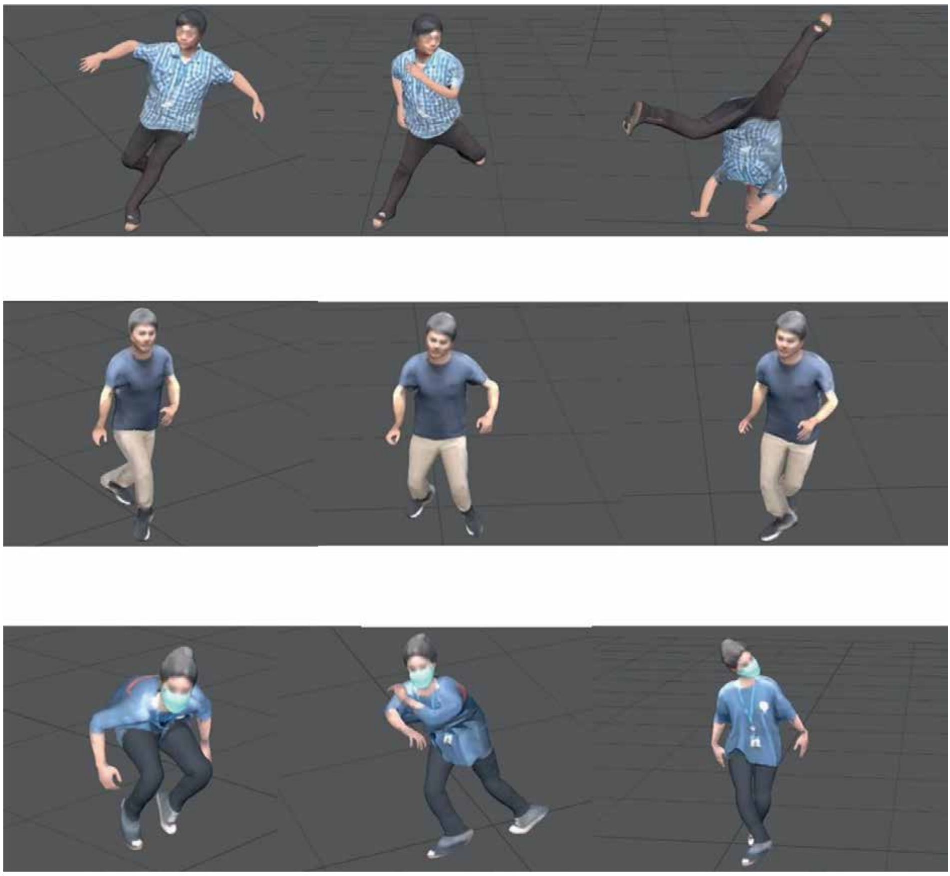


Figure 4.
Result of this chapter.

wide range of 3D games on the market. Compared to existing appearance generation methods [25, 26], this chapter focused on generating full-body models rather than just focusing on facial features. Additionally, the output format has been improved to enhance the versatility of the player-generated custom models for most games.

The model generation process in this chapter consists of three stages. The first stage involves generating the 3D character model using the parallel PIFu model, which captures front and back features, and then correcting the hand distortion issues using the HMR model. The second stage focuses on generating the character skeleton using the Pinocchio model, which conforms to the 3D model generated in the first stage. The third stage involves skeleton binding and animation generation, where the 3D model and its skeleton from the previous stages are bound and animated in the Maya environment. Finally, the model is outputted in the widely used FBX format for most 3D games. Although additional processing may be required for character animations, automatic generation is already possible for characters with skeletons. In the future, with the integration of a wider range of diverse training data, this method holds the potential to not only contribute to the advancement of the gaming industry but also facilitate progress and enhancement in the animation and film sectors, where extensive 3D modeling is a critical component. By leveraging this approach, these industries can benefit from improved efficiency and creative possibilities.

Conflict of interest

The authors declare no conflict of interest.

Author details

Hsuan-min Wang[1]*, Cheng-Lun Tsai[2] and Chuen-Tsai Sun[1]

1 National Yang Ming Chiao Tung University, Hsinchu, Taiwan,

2 National Chiao Tung University, Hsinchu, Taiwan

*Address all correspondence to: hmwang.cs04g@g2.nctu.edu.tw

References

[1] Loper M, Mahmood N, Romero J, Pons-Moll G, Black MJ. SMPL: A skinned multi-person linear model. ACM Transactions on Graphics (TOG). 2015;**34**(6):248

[2] Kanazawa A, Black MJ, Jacobs DW, Malik J. End-to-end recovery of human shape and pose. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Salt Lake City Utah. IEEE; 2018. pp. 7122-7131

[3] Alldieck T, Magnor M, Bhatnagar BL, Theobalt C, Moll GP. Learning to Reconstruct People in Clothing from a Single RGB Camera. arXiv preprint arXiv:1903.05885. 2019

[4] Huang Z, Li T, Chen W, Zhao Y, Xing J, LeGendre C, et al. Deep volumetric video from very sparse multi-view performance capture. In: Proceedings of the European Conference on Computer Vision (ECCV). Munich. Springer; 2018. pp. 336-354

[5] Saito S, Huang Z, Natsume R, Morishima S, Kanazawa A, Li H. PIFu: Pixel-aligned implicit function for high-resolution clothed human digitization. IEEE International Conference on Computer Vision (ICCV). 2019;**3**:2304-2314

[6] Sclaroff S, Pentland A. Generalized implicit functions for computer graphics. ACM SIGGRAPH Computer Graphics. 1991;**25**(4):247-250

[7] Chen Z, Zhang H. Learning Implicit Fields for Generative Shape Modeling. arXiv preprint arXiv:1812.02822. 2019

[8] Park JJ, Florence P, Straub J, Newcombe R, Lovegrove S. DeepSDF: Learning Continuous Signed Distance Functions for Shape Representation. arXiv preprint arXiv:1901.05103. 2019

[9] Mescheder L, Oechsle M, Niemeyer M, Nowozin S, Geiger A. Occupancy Networks: Learning 3D Reconstruction in Function Space. arXiv preprint arXiv:1812.03828. 2019

[10] Bogo F, Romero J, Loper M, Black MJ. Dynamic FAUST: Registering human bodies in motion. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Las Vegas Nevada. IEEE; 2016. pp. 2803-2812

[11] Liu P-C, Wu F-C, Ma W-C, Liang R-H, Ouhyoung M. Automatic animation skeleton construction using repulsive force field. In: Proceedings of the 11th Pacific Conference on Computer Graphics and Applications. Canmore. IEEE Computer Society; 2003. pp. 409-413

[12] Wade L, Parent RE. Automated generation of control skeletons for use in animation. The Visual Computer. 2002;**18**:97-110

[13] Lee T-C, Kashyap RL. Building skeleton model via 3-D medial surface axis thinning algorithms. Conference on Computer Vision, Graphics, and Image Processing (CVGIP). 1994;**56**(6):462-478

[14] Au OK-C, Tai C-L, Chu H-K, Or DC, Lee T-Y. Skeleton extraction by mesh contraction. ACM Transactions on Graphics. 2008;**27**(3):1-10

[15] Xu Z, Zhou Y, Kalogerakis E, Singh K. Predicting Animation Skeletons for 3D Articulated Models via Volumetric Nets. arXiv preprint arXiv:1908.08506. 2019

[16] Xu Z, Zhou Y, Kalogerakis E. RigNet: Neural rigging for articulated characters.

ACM Transactions on Graphics. 2020;**39**(4):58:1-58:14

[17] Renderpeople. [Online]. Availdable from: https://renderpeople.com/ [Accessed: June 01, 2023-06-01]

[18] Sloan P-P, Kautz J, Snyder J. Precomputed radiance transfer for real-time rendering in dynamic, low-frequency lighting environments. ACM Transactions on Graphics. 2020;**21**(3):527-536

[19] Johnson S, Everingham M. Clustered pose and nonlinear appearance models for human pose estimation. British Machine Vision Conference (BMVC). 2010;**2**(4):5

[20] Andriluka M, Pishchulin L, Gehler P, Schiele B. 2D Human Pose Estimation: New Benchmark and State of the Art Analysis. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Columbus. IEEE; 2014. pp. 3686-3693

[21] Lin T-Y, Maire M, Belongie S, Bourdev L, Girshick R, Hays J. Microsoft COCO: Common Objects in Context. arXiv preprint arXiv:1405.0312. 2015

[22] Ionescu C, Papava D, Olaru V, Sminchisescu C. Human3.6M: Large scale datasets and predictive methods for 3D human sensing in natural environments. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2014;**36**(7):1325-1339

[23] Mehta D, Sridhar S, Sotnychenko O, Rhodin H, Shafiei M, Seidel H-P, et al. VNect: Real-time 3D Human Pose Estimation with a Single RGB Camera. arXiv preprint arXiv:1705.01583. 2017

[24] Baran I, Popovic´ J. Penalty Functions for Automatic Rigging and Animation of 3D Characters. In: Proceedings of the 2007 SIGGRAPH Conference. San Diego. ACM SIGGRAPH. 2007;**26**(3)

[25] Shi T, Yuan Y, Fan C, Zou Z, Shi Z, Liu Y. Face-to-Parameter Translation for Game Character Auto-Creation. arXiv preprint arXiv:1909.01064. 2019

[26] Lin J, Yuan Y, Zou Z. MeInGame: Create a Game Character Face from a Single Portrait. arXiv preprint arXiv:2102.02371. 2021

Chapter 5

The Role of Game Engines in Game Development and Teaching

Branislav Sobota and Emília Pietriková

Abstract

This chapter explores the essential role of game and graphics rendering engines in creating computer games and their applications in education. Starting with a brief discussion on the importance of visualization in game development, we analyze popular game engines, including CryEngine, Irrlicht, Unreal Engine, Unity, and Godot, and graphics rendering engines (cores), including OpenSceneGraph (OSG) and Object-Oriented Graphics Rendering Engine (OGRE). Each engine's unique features and pedagogical potentials are discussed, with emphasis on Unity, Godot, and Unreal Engine's interactive and step-by-step tutorials. This analysis reflects the adaptability of game engines to different teaching styles and provides insights into their effectiveness as tools for teaching game development and computer science. The chapter aims to highlight game engines' role in game creation and education by exploring these topics.

Keywords: game engines, teaching game development, computer science learning, visualization, game engines tutorial

1. Introduction

Computer games and, of course, especially their development, are a significant part of computer science. Modification or change of some procedures in game development is necessary due to modern technologies. In this sense, the development of technical means, especially graphics adapters, is advancing, for example, the possibilities of computer graphics as an essential component of computer games. On the other hand, the opportunities and problems of graphics libraries and engines define the need and maximization of graphics adapters' use. This aspect explains a significant synergistic connection between computer science and computer games. This connection is also reflected in the basic architecture of the currently standard gaming computer system (**Figure 1**).

As a person perceives up to 80% of information by sight, it is understandable that the visualization subsystem is dominant in this case. And the central phenomenon of the visualization subsystem is the use of computer graphics. Its usage can be seen at distinct levels of the hierarchy (**Figure 2**): starting with graphics software at the machine code level and ending with complex graphics engines. To this, it is necessary to add the graphic options of higher programming languages/browsers use,

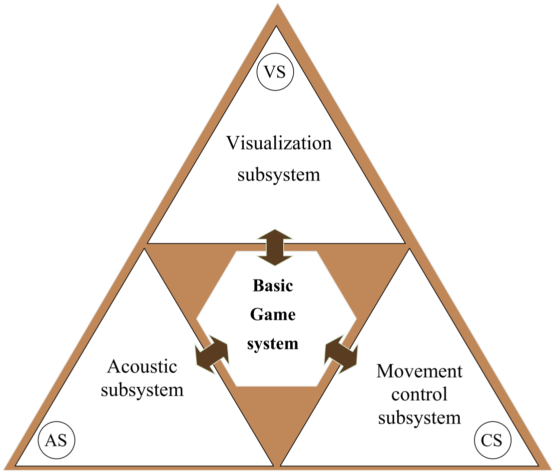


Figure 1.
Basic game system architecture.

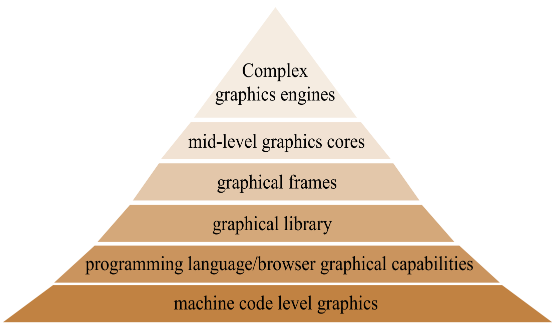


Figure 2.
The levels of computer graphics usage hierarchy.

graphic libraries (e.g., OpenGL[1] (Vulcan[2]), (webGL[3]), graphic frames (e.g., ThreeJS[4], BabylonJS[5]), or medium-level graphic cores (e.g., OpenScenegraph[6], A-frame[7]).

It is crucial to present this relationship in this way, also when teaching IT (Information Technologies) students. Including all the mentioned levels in the teaching introduces individual levels to students and shows the increase in the possibility of using computing technology in this area. In addition, increasingly higher levels of hierarchy expand the circle of users, especially game developers, beyond the exact group of IT specialists. In this context, virtual reality (VR) and its advanced technologies also play an essential role.

1 OpenGL, home page: [cit. 2023-03], URL: http://www.opengl.org/
2 Vulkan, home page: [cit. 2023-03], URL: https://www.vulkan.org/
3 webGL, home page: [cit. 2023-03], URL: https://get.webgl.org/
4 ThreeJS, home page: [cit. 2023-04], URL: https://threejs.org/
5 BabylonJS, home page: [cit. 2023-04], URL: https://www.babylonjs.com/
6 OpenSceneGraph, home page: [cit. 2023-03], URL: http://www.openscenegraph.org/
7 A-Frame, home page: [cit. 2023-04] URL: https://aframe.io/

When teaching game design and development, introducing game theory can enhance students' understanding of game mechanics, dynamics, esthetics, balance, etc. Utilizing game engines in the classroom provides an excellent platform for practically applying game theory concepts. For instance, students can use game engines for prototyping and testing different game mechanics while observing the result of various strategies and interactions and effectively experiencing game theory in action. While this chapter focuses on the role of game engines in teaching game design and development, the interplay between these engines and game theory underpins the educational potential of these tools. As students learn to use game engines, they inherently deal with game theory concepts, bringing abstract mathematical models to life.

That is why the Department of Computer and Informatics of the Faculty of Electrical Engineering and Informatics (FEEI) at the Technical University of Košice has been running the Design and Development of Computer Games course and related subjects (e.g., Computer Graphics or Virtual Reality Systems) for several years. As we assume that IT (Information Technologies) students are potentially future creators/programmers of computer games, they must have at least a basic understanding of game theory to be successful in the implementation. Therefore, as a part of the educational process, students become familiar with various methodologies and technologies related to the design and development of computer games, such as high concept, market research, storytelling, prototyping, balancing rules, Mechanics-Dynamics-Esthetics (MDA), or iterative game development.

Game engines are integral to various domains, including game development, VR, and computer science. The creation of computer games is a sophisticated process that requires artistic vision and robust technology to bring that vision to life. The essential tools used in this process are game engines, handling many necessary elements, such as rendering graphics, processing physics, and managing inputs. In this chapter, we explore the crucial role of game engines in developing computer games and learning.

Following this, we analyze several popular visualization cores and game engines, presenting their unique features, strengths, and weaknesses. In the latter part of the chapter, we shift our focus to the educational implications. We illustrate how different approaches can influence learners through practical tutorials in Unity[26], Godot[21], and Unreal Engine[20]. Our exploration underscores the value of practical, applied learning in equipping students with industry-relevant skills. We aim to highlight the role of game engines in game creation and education.

We will also present the outputs of some activities recorded during the educational process with students in the third year of the Informatics bachelor's degree in a full-time study form. We attach significant importance to the fact that these are real procedures and activities processed over several years and can thus provide a basis for further studies. Therefore, we believe the experiences presented in this chapter will find their readers.

2. Visualization subsystems in games

In general, the purpose of the game's graphics or visualization subsystem, in particular, is to support the loading and rendering of the scene for visualization. It defines the data format for the scene and handles user input. There are additional features in systems such as special effects and physics support, and many computer features include an integrated environment for scene development and scripting.

The systems with extended functionality are also called 3D (three-dimensional) game engines if their primary purpose is to support game development.

Among the functions that can be associated with the visualization subsystem of the game concerning its implementation are:

- *License*. The license must allow the software to be used for commercial purposes.
- *Software extensibility and source code availability*. There should be an option to modify the selected candidate using its application programming interface (API), but the best option would be to have its source code available for customization.
- *Quality of documentation*. Documentation should be complete, understandable, and up-to-date.
- *Support for standard 3D modeling formats*. They determine the possibilities of importing 3D models created using other software or scanned using 3D scanners.
- *Support for the latest 3D graphics features*. It is vital for creating the most realistic video output possible.
- *Programming languages available for scripting*. Scripting, or so-called soft coding, is a way to program visualization controls without compiling. In the case of commercial products, this is often the only way to communicate with the core at the programmatic level.
- *Support for advanced virtual reality hardware solutions*. In this sense, focusing more on supporting stereoscopic video output and sensing user movements is necessary.
- *Availability of development tools for scene creation and scripting*. The availability of such a development tool (editor or integrated development environment (IDE)) can significantly reduce the time required to prepare a visualization scene.
- *Parallelization options*. Rendering high-quality video output requires a powerful computer system. Within game systems, we often need to display more than one output; for example, two images are necessary for stereoscopic display. In addition, immersive solutions often contain multiple displays (for example, front, back, left, right, top, and bottom in the case of a virtual cave). To maintain high quality, the images must be rendered in parallel.

In addition to these parameters, prototyping and game testing are also important. If we talk about prototyping, we recognize three types of game prototypes [1]:

- *Software.* Prepared in a game engine like Unity.
- *Physical.* Physical interaction and activity with the playtester.
- *Paper.* Board game, the most common type.

In the case of testing, the literature states that four types of testing are crucial to the design and development of computer games [2, 3]: Focus Groups, Quality Assurance (QA) Testing, Usability Testing, and Playtesting.

Next, we will provide an overview of some available visualization cores or complex game engines, both free solutions and commercial ones.

2.1 Visualizing and manipulating the game world

Something has already been hinted at visualization earlier, and of course, much more will be written about it, as it is currently a key part of a computer game. Therefore, let us mention here that, from a technical point of view, visualization is possible using various technologies. Manipulation of objects is also possible here. For interactive manipulation of objects in the game, it is possible to use classic input devices like a keyboard and mouse or various position sensors. The overall flow of data between individual structural elements from the point of view of working with the game world is depicted in **Figure 3**.

The visualization subsystem of the game provides the central part of the information for the player in the game environment. This subsystem calculates the image for the player in the game world. For visualization, the visualization core mentioned above is often used in games. The visualization core is usually based on the serial use of information, and currently, the graphics processing unit (GPU) (graphics processor at the hardware level) services are used. On the input side, there is a model of the game world, and on the output side, there are correctly colored pixels on the display. Between these two stages, other stages partially change the input information. The main idea for implementing the game visualization subsystem is to minimize the time used in each stage for the real-time system, which is hugely often needed in games. In multiprocessor systems, including General-Purpose Graphics Processing Unit (GPGPU) technology, it is possible to apply some parallel stages, representing a significant speedup.

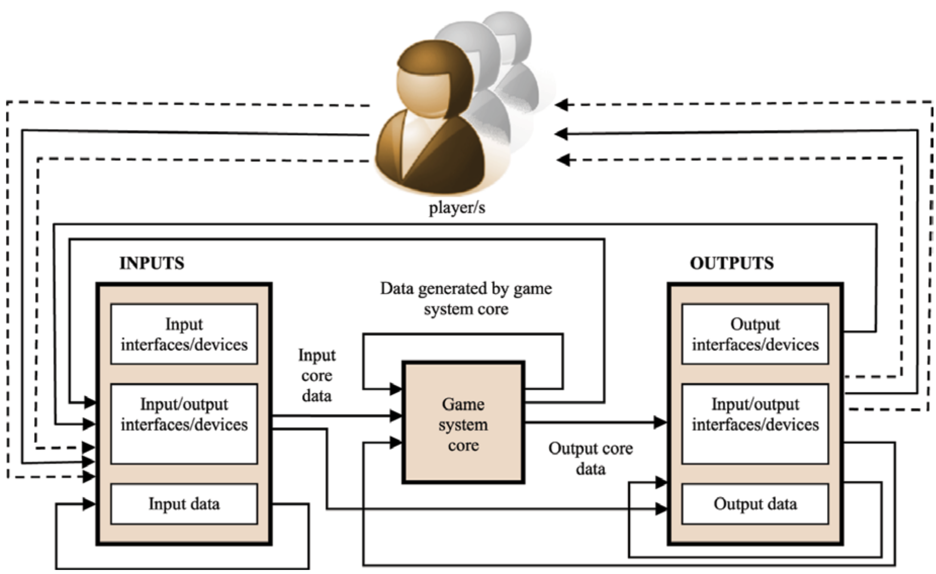


Figure 3.
Basic data flow in a computer game.

Complex game engines, at the highest level of the hierarchy (**Figure 2**), often support visual programming in the form of WYSIWYG (What you see is what you get) technologies (e.g., Blueprints for Unreal Engine). WYSIWYG (What you see is what you get) and other interactive tools thus simplify the entire development of the game and, in principle, bring this process closer even to people without an exact IT education. For example, the following features are available:

- Integrated graphic editors of the scene and dynamics.
- Integration of programming and scripting languages.
- Integration of support for different platforms and operating systems.
- Integration of support for various technical means, including network communication or parallelization of calculation.

3. Visualization cores and game engines

In this section, we will provide an overview of some available visualization cores and complex game engines, both free solutions and commercial ones, which are used or have been used as part of the courses at the authors' affiliation.

Of the freely available solutions, OpenSceneGraph[6] is interesting, which is not the easiest to understand and integrate but supports parallel calculations and modern VR technologies. Many implementations in the world form an excellent knowledge base. These factors currently make it the favorite in this category for the need to integrate graphics systems into game solutions unrestrictedly.

The commercial solutions mentioned next support the most modern imaging techniques and provide limited options for integrating modern input 3D interfaces (for example, based on virtual reality). However, compared to freely available solutions, they enable the quick creation of scenes thanks to integrated editors, which are very well documented. On the other hand, the limited possibility of modifications, only sometimes available source code (or at least not at an affordable price), and no or limited support for parallel computing, could be a limitation in distributed gaming environments.

3.1 OpenSceneGraph (VulkanSceneGraph project)

OpenSceneGraph[6] (OSG, **Figure 4**) is a freely available high-performance 3D visualization core written in the C++ programming language (not to be confused with OpenSG[8], which is no longer in development and represents an entirely different scene graph API, somewhat similar to OpenGL Performer). It uses the OpenGL[1] library (the VulkanSceneGraph Project[9] is available for the Vulcan library[2]). It is multiplatform and can be used on MS Windows, OS X, and GNU/Linux platforms. Since version 3.0.0 (3.6.5 currently), it is available also for mobile platforms, Android and iPhone Operating System (iOS). It is also used as the basis for several virtual reality solutions, including JuggLua VR, CalVR, and Vrecko.

[8] OpenSG, home page: [cit. 2022-06], URL: https://sourceforge.net/projects/opensg/

[9] VulcanSceneGraph, home page: [cit. 2023-03], URL: https://vsg-dev.github.io/vsg-dev.io/

Figure 4.
Visualization samples based on OpenSceneGraph.

Figure 5.
Visualization samples based on object-oriented graphics rendering engine (OGRE).

Scenes are represented by a data structure called a scene graph, which organizes a logical and often spatial representation of the scene. The documentation is not OSG's strong point. It contains several user and programmer guides. Most used 3D formats are supported through plug-ins that are part of the OSG core distribution. OSG effectively implements the best graphic techniques, such as early elimination of the primitives based on overlap and visibility, progressive simplification of geometric detail of models, sorting of display states, and so-called particle systems. Support for graphics chain programming (shader) is implemented based on OpenGL. Of the analyzed solutions, OSG has the longest history of use in games or visualization systems. Support for stereoscopic visualizations and peripheral input devices has been implemented. There are several extensions, for example, *osgTerrain* (terrain rendering), *osgAnimation* (character and rigid body animation), *osgManipulator* (3D interactive controls), *osgShadow* (shadow framework), *osgParticle* (for particle systems), *osgFX* (special effects framework), or *osgVolume* (high-quality volume rendering (with Dicom plug-in for support of medical datasets)). OSG directly supports parallel computing because the scene graph supports multiple graphics contexts and stores graph states in fast local memory. Another option for parallelization is using the program *Equalizer* tool[10].

3.2 Ogre

OGRE[11] (**Figure 5**) is a scene-oriented 3D core. It is created in C++ object programming, and the abbreviation is made from the Object-Oriented Graphics

[10] Equalizer, home page: [cit. 2023-01], URL: http://www.equalizergraphics.com/

[11] Ogre3D, home page: [cit. 2023-01], URL: http://www.ogre3d.org/

Rendering Engine. It is a pure displaying core, so support for playing sounds, physics, and other features is at the plug-in level. Its expandability is guaranteed by a highly modular architecture based on plug-in modules. It supports Direct X[12], OpenGL[1], and Vulcan[2], and it is distributable not only to MS Windows, Mac OS X, and Linux but also to Android and iOS platforms. Its highly rated feature is excellent support for animations based on skeleton models.

Object-Oriented Graphics Rendering Engine (OGRE) is freely redistributable with open-source code, so price and licensing are no issues. It is very well documented, and in addition to a rich set of freely available sources, there are two books [4, 5]. OGRE directly supports only its model format, but fortunately, there are export libraries in plug-in models to several 3D editors. The support of advanced graphic standards is at a reasonable level. The high-level shader language (HLSL) supports graphics string programming. The scripting is not directly available; however, some extensions add this capability. The support for stereoscopy is again provided by a unique project called *Stereo Vision Manager*. The VRPN[13] (Virtual Reality Peripheral Network) library can support advanced 3D interfaces. Furthermore, some extensions make it easier to integrate this solution. OGRE does not include direct support for parallelization, but there are several attempts for this extension, again as in the case of OpenSceneGraph[6] using the *Equalizer* tool[10].

3.3 CryEngine

CryENGINE[14] (**Figure 6**) is a professional 3D game engine. It has been used in the creation of several highly rated game titles. In addition to the game world, virtual training for American army soldiers is also based on it. It is available for MS Windows and Xbox, Playstation, and Wii game platforms. It also has direct support for playing sound effects and physics.

The users can use this 3D engine within the academic sector without fees. For commercial use, it is necessary to pay a fee, which depends on the purpose of the developed title, and it is determined based on communication with the manufacturer. The core is written in C++ and can be extended via API or scripting. The documentation is of a high standard, as is customary with commercial products. It supports only

Figure 6.
Visualization samples based on CryENGINE.

[12] DirectX, home download page: [cit. 2023-02], URL: https://www.microsoft.com/en-us/download/details.aspx?id=35

[13] VRPN, home page: [cit. 2023-03], URL: http://www.cs.unc.edu/Research/vrpn/

[14] CryENGINE, home page: [cit. 2023-02]. URL http://mycryengine.com/

the internal format of 3D models but for the most used 3D editors, such as 3D Studio Max or Blender, official exporters, in the form of plug-ins and extensions. This core, built on DirectX[12] in all its versions, has always supported the current state-of-the-art display techniques. There are two scripting methods: either using the LUA[15] language or a visual scripting editor called *FlowGraph*. The stereoscopic display is directly supported. There must be a known implementation of support for modern 3D interfaces yet. A sophisticated editor called Sandbox is used to create the scenes. The parallelization is not directly supported, but the synchronization of multiple computers running this core can, in principle, be implemented (this was done in the project mentioned in [6]).

3.4 Irrlicht

Irrlicht[16] (**Figure 7**) is freely distributable with open-source C++ code. It runs on multiple platforms. These are primarily MS Windows, Mac OS X, and Linux, but other distributions exist. It has been a stable engine in the works for 20 years. The engine was created in 2003 by a single developer, Nikolaus Gebhardt. Currently, the development team consists of approximately 10 members. Both DirectX[12] and OpenGL[1] can be used for display, as well as internally developed software rendering. One of the main advantages of this engine is no requirement for the installation of additional third-party libraries, making it quite easy to install and run.

Since Irrlicht is freely distributable and can be used commercially, the source files may be modified, extended, and redistributed without a charge. It is well documented, including a description of the programming API, and it includes tutorials with examples for straightforward and advanced use cases. Two books also deal with its description and examples [7, 8]. Irrlicht has the broadest support for input 3D formats compared to other solutions. Importing outputs from the editors, such as 3D Studio Max, Maya, and even forms of different 3D engines, e.g., OGRE, is possible. The scripting has yet to be officially supported, but again there are solutions created by the community that add this functionality for languages such as Ruby[17], Python[18], Lua[15], or Perl[19]. However, some of them are problematic, and support for their

Figure 7.
Visualization samples based on IrrLicht.

[15] LUA, the programming language, home page: [cit. 2023-01]. URL: http://www.lua.org/

[16] Irrlicht, home page: [cit. 2022-09], URL: https://irrlicht.sourceforge.io/

[17] Ruby, home page: [cit. 2023-01], URL: http://www.ruby-lang.org/en/

[18] Python programming Language, home page: [cit. 2023-01], URL: http://www.python.org/

[19] Perl, programming language, home page: [cit. 2023-01], URL: http://www.perl.org/

development has already ended. Irrlicht includes a free, extensible 3D editor called *irrEdit*. The parallelization is not directly supported but can be implemented in principle as the source files are available.

3.5 Unreal Engine

Unreal Engine[20] (UE, now UE5 (**Figure 8**)) was developed by Epic Games more than 24 years ago as a first-person shooter (FPS) system. Unreal Engine has evolved to support many other genres, including diverse types of games and even linear video production, such as animated stories and 3D VR walkthroughs. It should come as no surprise that, for example, instructional designers are using it to create enterprise virtual reality applications. UE is a commercial solution, the use of which is the same as that of Unity. Unreal Engine supports the deployment of projects on multiple platforms, including MS Windows PC, Mac OS X, iOS, Android, AR, VR, Linux, and HTML5, as well as on head-mounted displays (HMDs) and game consoles. The Unreal Engine editor runs on MS Windows, OS X, and Linux. Due to its more intensive use, we will describe it in more detail.

When programming in UE5, there are two main ways to implement logic into games/experiences: the Blueprints visual scripting language and the more traditional C++ programming language. Unlike *Blueprints*, C++ can be a bit obscure as it can take some time to learn the required syntax, but C++ provides better access to some of the engine's hidden features. *Blueprints* also offer an easier way to migrate work from one project to another. Within UE5, several additional tools and toolkits are available for the tasks such as animation, creating and modifying game environments, and multiuser (collaborative) editing.

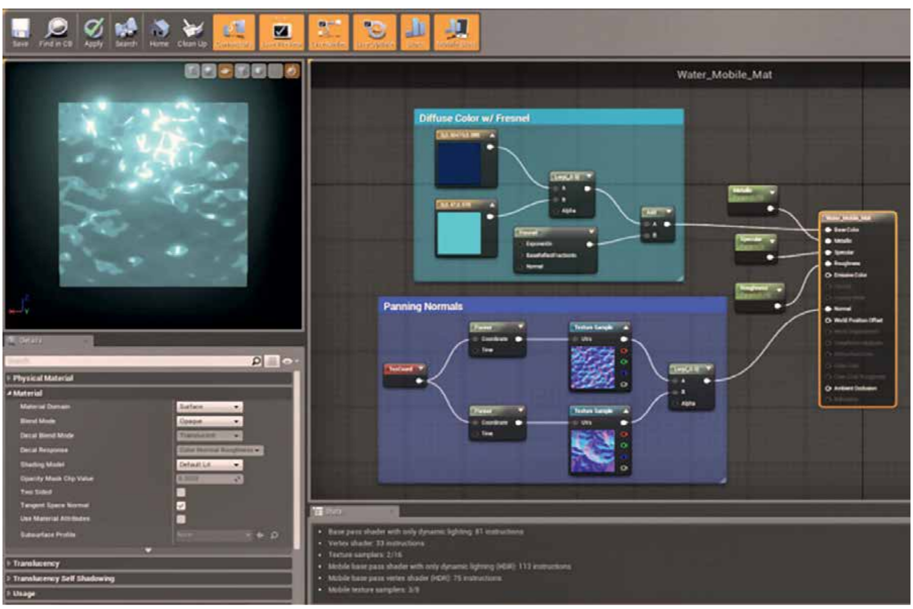


Figure 8.
Material creation in the Unreal Engine.

[20] Unreal Engine, home page: [cit. 2023-03], URL: https://www.unrealengine.com/

A UASSET file is an asset file used by the Unreal Editor, a game-level editor, part of the Unreal Engine. It contains assets such as Game Levels, Particle Systems, Blueprint Scripts, Materials, Movie Sequences, Textures, Static Meshes, Skeleton Meshes, Light Profiles, or soundtracks used in the game. All objects/models from the Unreal Engine environment are available as a UASSET file.

Users can create objects/models outside the Unreal editor using tools such as Maya, 3D Studio Max, or Photoshop. They then import these assets into Unreal Editor, which converts the objects/models into UASSET files.

In addition to excellent visualization capabilities, Unreal Engine 5 provides functions to support multiple players/users (client-server model), especially in multiplayer games. Overall, Unreal Engine includes a robust networking framework that is also used by some of the most popular online games in the world to help streamline the process. In multiplayer/user mode, Unreal Engine uses the already-mentioned client-server model. One device on the network acts as a server and hosts a multiplayer game session, while all other players' devices connect to the server as clients. The server shares game state information with each connected client and provides a means for them to communicate. The server, as the host of the game, has only the actual and authoritative state. In other words, multiplayer games take place on the server. Each client remotely controls their game instance to the server (Pawn) and sends procedure calls to perform game actions on the client. However, the server does not transmit visual information directly to client monitors. Instead, the server replicates game state information to each client, telling it what actors exist, how those actors should behave, and what values various variables should have. Each client then uses this information for the visualization itself.

Unreal Engine is currently free to download and use. One must pay after the game or app is released. The fee is from the profit after earning the first $3000 per product per quarter. It is also possible to download the complete UE source code and modify it according to the needs of the project being created. As mentioned earlier, users can write new classes and parts of the user interface in C++. UE Blueprints are usable for scripting game logic using nodes.

As part of solving various projects at the authors' home department, UE was used, for example, to create a virtual-reality therapeutic system for paretic patients [9] (**Figure 9**).

3.6 Godot engine

The Godot engine[21] (**Figure 10**) is an open-source project under the MIT license, making it available for free download, use for personal and commercial purposes, modification, repair, or extension of its source code. This is also one of the reasons why this engine is a viable choice for students and beginners.

The Godot engine is cross-platform, allowing the game's creation for Windows, Linux, MacOS, mobile phones, web browsers, and consoles, although the support here is weaker. Both 32- and 64-bit versions of Windows and Linux or MacOS are supported. Godot supports two-dimensional (2D) and 3D game development, but it excels especially in 2D, as mentioned by Dealessandri[22]. A 3D game creation is possible but more complicated.

Multiple programming languages are supported. First, the GDScript scripting language is specially created for the engine's needs. The syntax reminds us of Python;

---

[21] Godot Engine, home page: [cit. 2023-02], URL: https://godotengine.org/

[22] URL: https://www.gamesindustry.biz/articles/2020-04-14-what-is-the-best-game-engineis-godot-right-for-you/ [cit. 2023-05].

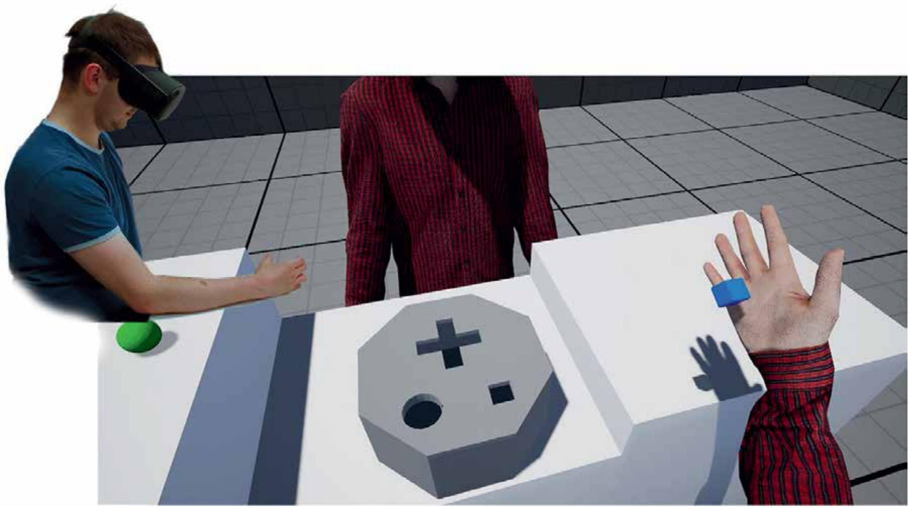


Figure 9.
Example of the Unreal Engine (UE) use within a Virtual Reality (VR) training system for paretic patients.

Figure 10.
Game samples based on the Godot engine.

however, it is closely adapted to the engine, increasing its optimization. Its main advantage is the reduction of development complexity. In addition to GDScript, Godot supports C#, C++, and VisualScript.

Its developers try to keep the core of its functionality small. The effort is to add functionalities as plug-ins. One of the reasons is to have as minimal maintenance as possible for code that constantly needs to be tested. Furthermore, it makes the compilation of games faster. Thanks to its smaller and cleaner code, this makes contributing to the engine project easier. And thus, the size of the binaries of the editor and the exported projects is kept small.

The Garden Path[23] is an exciting game created in the Godot engine. It is a 2D top-down adventure game. Another game is Resolutiion[24]. It is an esthetically pleasing 2D top-down action adventure with a pixel art style. The Human Diaspora game[25] is an excellent example of how 3D projects can be created in the engine. It is a 3D First-Person Shooter (FPS), i.e., a shooter with a first-person view. The Dauphin game is also enjoyable. It is a 2D Role-Playing Game (RPG), i.e., a game in which the player

[23] URL: https://store.steampowered.com/app/1638500/The_Garden_Path/ [cit. 2023-05].

[24] URL: https://store.steampowered.com/app/975150/Resolutiion/ [cit. 2023-05].

[25] URL: https://store.steampowered.com/app/1395420/Human_Diaspora/ [cit. 2023-05].

stands in the role of the character he is playing and gradually builds him up and increases his abilities.

One very high-quality and well-crafted instruction source is the GDQuest channel on YouTube. This channel also has a webpage offering paid courses in Godot. We recommend [10] from the books.

3.7 Unity Engine

Unity[26] (**Figure 11**) is a complex game engine. It represents a commercial solution. It is among the most popular game engines due to its ease of use and the wide range of functionalities it offers. Unity is a cross-platform game engine developed by Unity Technologies. As mentioned, it ranks among the most popular game engines, as it is suitable for beginners and offers a wide range of tools for 2D and 3D projects. Among these tools, we can include collision detection, 3D rendering, effects, physics simulation, or a simple editor. The main advantages of Unity include a large community of game developers, a lot of tutorials, and easy export of the game to multiple platforms. Unity is especially popular with smaller and medium-sized projects, as it allows for rapid prototyping and works on a system of adding components to game objects, thanks to which it is possible to lay out and change the scene quickly. Supported platforms include, for example, MS Windows, Mac OS, Linux, Xbox, Wii, and Playstation, as well as Android and iOS mobile solutions. Depending on the target platform, Unity supports both DirectX[12] and OpenGL[1].

Unity has a graphical environment adapted for easy game creation like other game engines. It consists of the Editor (**Figure 11** right), consisting of several windows and panels, each with unique functionality. In the image below, we can see the default layout of these windows. Basic windows include:

- *Hierarchy*—It serves to display all objects on the scene.
- *Inspector*—We can edit and add properties of the selected object.
- *Game view and scene*—It serves to simulate the game and edit the scene.
- *Project*—It contains all the necessary files that we use in the project.

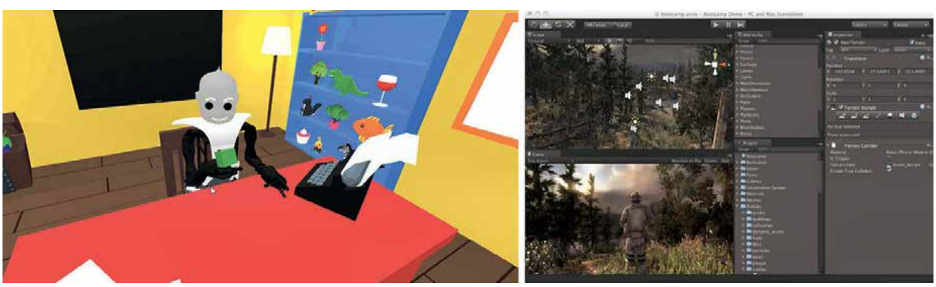


Figure 11.
Example of the simple game in UNITY and visualization in the integrated Unity editor.

[26] Unity Game Engine, home page: [cit. 2023-03]. URL: https://unity.com/

Essential elements of the engine include prefabs, colliders, physics, an audio system, and animations. Prefabs are used as game objects with preset properties and components, thanks to which we can use them in several scenes and easily duplicate or instantiate them. Colliders are used to detect collisions between objects. Unity also includes a physics engine that simulates gravity, acceleration, and other physical phenomena. These tools make it easy for game developers to create prototypes and focus on developing specific parts of the game.

From an educational perspective, several tutorials explain how to work with Unity. Some points are directly taken from this engine's developers or professionals who create various video tutorials.

3.8 Teaching and learning with game engines

Game engines not only offer students different pathways for learning but also provide educators with various teaching methods. Each game engine's unique functionality and design can support different teaching styles [11, 12].

CryEngine's flowgraph system offers a graphical scripting interface, which could support a visual or demonstrative teaching style. Educators can visually illustrate game mechanics [13], providing an intuitive way for students to understand game logic without the necessity of text-based coding. On the other hand, Irrlicht, as a lightweight and easy-to-use graphics engine, could support a direct teaching style where educators focus on teaching the underlying principles of 3D graphics without the complexity or distraction of more extensive features [14]. Its simplicity could be especially beneficial when teaching beginners or focusing on foundational concepts.

With its advanced features, Unreal Engine is well suited to a teaching style where the educator imparts detailed knowledge on complex topics like advanced graphics, physics simulation, or artificial intelligence (AI) programming. Its powerful visual scripting system, Blueprint, also provides an avenue for teaching programming concepts more visually and intuitively, supporting diverse learners [15]. For learners aiming to grasp the technicalities of game physics and lighting, Unreal provides a highly illustrative platform.

Godot's unique scene system lends itself to a more systematic teaching approach. Educators can build lessons around the node and scene structure, teaching individual concepts and showing how they fit into a larger whole. This modular approach could support a scaffolded teaching style, where each new concept builds on the previous one [16]. The engine's node and scene structure encourage a modular approach to game design, which can facilitate a deep understanding of object-oriented principles, design patterns, and aspects of game development like the parallax effect.

Unity's user-friendly interface and real-time feedback can support a more facilitative teaching style, allowing educators to guide students through hands-on experimentation. The immediacy of seeing the effects of code changes in the game world provides tangible feedback that reinforces learning. Unity's user-friendly interface and robust community support make it accessible to beginners [17]. Unity's strong community and extensive online resources also provide ample material for independent study, supporting a flipped classroom approach where students explore content outside of class and use classroom time for discussion and application.

Game engines offer educators a range of possibilities to align with their preferred teaching styles. Educators can create engaging, effective learning experiences that

Engine	Teaching approach	Learning style	Suitable for
OSG	Detailed, methodological	Analytical learners	Advanced course, graphics programing
OGRE	Problem-based	Practical, hands-on learners	Object-oriented programming (OOP) teaching, intermediate to advanced courses
CryEngine	Project-based	Visual and kinesthetic learners	Advanced courses, high-fidelity game development
Irrlicht	Direct instruction	Systematic learners	Beginner to intermediate courses, low-poly game development
Unreal Engine	Demonstrative, project-based	Visual and kinesthetic learners	Intermediate to advanced courses, high-fidelity game development
Godot	Step-by-step, modular	Sequential learners	Beginner to advanced courses, 2D and 3D game development
Unity	Interactive, exploratory	Active, hands-on learners	Beginner to advanced courses, rapid prototyping, VR development

Table 1.
A comparison of teaching and learning approaches in game engines.

resonate with their students by choosing an engine that complements their approach. **Table 1** outlines each engine's general teaching approaches and learning styles[27].

Table 1 illustrates the diverse teaching opportunities presented by different game engines, making them adaptable to various teaching styles and educational contexts. In the following section, we will delve deeper into the educational application of game engines. We will present different tutorials, each adopting various teaching approaches using Unity, Godot, and Unreal Engine. These examples will demonstrate how these engines can be tailored to facilitate different teaching styles, cater to varied learner preferences, and ultimately foster effective learning experiences in game design and development.

4. Educational experience with game engines

Our institution focuses on teaching game creation within one academic course and partially in the next four courses. The lectures are devoted to the most critical aspects related to the design and development of games. The laboratories focus on technology and teamwork. As a part of these laboratories, we decided to show students work in three popular engines: Unity, Unreal, and Godot, working with each differently.

For each tutorial, we dedicated two laboratories. During the first laboratory, we introduced the game engine and started with the tutorial, helping students with troubleshooting. This initial introduction was crucial in assisting students to navigate the engine environment and understand the basics. During the second laboratory, we let students work on the tutorials independently. Then, the students finished their work at home.

[27] Please note that this table serves as a guideline rather than a strict rule, as these engines' versatility allows various teaching methodologies to be employed based on the specific course objectives, student backgrounds, and teacher preferences.

4.1 Pedagogical value of game engines

Game engines transform the learning process into a more engaging and practical experience by offering students a platform to visualize and interact with abstract concepts. They provide a realistic context for students to grasp abstract concepts in game development and computer science, e.g., mirroring class objects with game objects or object manipulation through different components.

Game development calls for the application of problem-solving skills. Game engines challenge students to design, implement, and troubleshoot, reinforcing their theoretical knowledge with practical application. Game development offers students exposure to various fields, e.g., computer graphics, physics simulations, artificial intelligence, and more. In the case of Unreal Engine, students can experiment with AI behavior, graphical effects, or physics-based gameplay.

Game engines hold immense potential as pedagogical game development and computer science tools. As technology advances, they make learning an interactive, engaging experience. Educators can equip students with versatile skills beyond traditional computer science curricula.

4.2 Interactive tutorial in Unity Engine

For this tutorial[28], we decided to be inspired by the Unity Technologies company, which in 2017 decided to include an interactive tutorial directly in their game engine. The manual contained tasks in text form, updated after completion. It highlighted specific windows in the editor for the user to focus on. Unfortunately, due to the constant influx of new versions of Unity, this tutorial did not have time to adapt to the current versions, so standard tutorials replaced it.

Our goal was to create a manual that would eliminate the shortcomings of already existing manuals while preserving the parts that work. That is why we have set the following goals:

- *Guiding users* along a specific path using the level system, thanks to which we can refer to already-explained principles and thus avoid unnecessary repetition.

- *Eliminating the monotony* of regular instructions by adding multiple forms of explanation of functionalities in videos, text descriptions, and practical tasks.

- *Enabling direct interaction* with the environment by creating an integrated tutorial directly into Unity and adding tasks to practice the explained parts of the tutorial.

- *Create a motivating, fun environment* by adding various gamification elements, such as avatars, collecting money, and unlocking levels.

- *Explanation of the fundamental Unity functionalities* by creating six levels focused on the essential parts of the engine.

[28] Interactive tutorial in Unity [cit. 2023-06]. URL: https://kurzy.kpi.fei.tuke.sk/gamedev/en/book/unity.html

DOI: http://dx.doi.org/10.5772/intechopen.1002257

During the technical design of classes and components in Unity, we created the tutorial's basic structure as an easily scalable system of sections. Each section consists of subsections or steps: text, test, or video. Text steps create descriptions in text form, videos are linked to a video player, and tests are linked to scripts verifying student knowledge. Navigating the sections works by switching steps using the forward and backward buttons. The main manager is the manager who keeps the information about the current section and step, which is displayed and is responsible for switching steps.

The main idea of the tutorial was to create a comprehensive overview of Unity's basic functionalities, which will guide the user. Therefore, we designed six levels, divided into smaller parts dealing with specific Unity areas. The levels follow each other, while their last part follows the beginning of the next level. When creating the content, we tried to devote enough time to the structure of the sections, especially their alignment. It was necessary to find out which parts were related to each other to avoid confusion when going through the instructions.

We designed the environment to provide seamless navigation on screens and be user-friendly. The main screens include an introduction, the main menu, the shop, and a screen with instructions—text descriptions, video, and tasks.

The central part of the menu comprises the levels, which, when clicked, start a section with a given topic. They contain a button component that, when pressed, activates the corresponding section of the tutorial, and loads the last screen of the section. After going through the instructions, the locked next level will unlock, and the level will be set as completed.

We also created a store as a gamification element of motivation and reward, as users can unlock a new avatar for money earned by passing levels (**Figure 12**).

Steps are the smallest element in the tree of sections and thus are its leaves. There are three types of steps:

- *Text.* It is the simplest type of step, as it contains only a text string in the given step and a URL (uniform resource locator) link to the documentation. It is used for introductory descriptions of functionalities.

- *Video.* A video clip representing a detailed guide to the given part, playable within the tutorial.

- *Test/assignment.* It serves to verify students' knowledge through a short exercise. It contains a reference to the verification script, the text displayed at the beginning of the test and in the help, and a link to the video clip played in the help. The user must complete the task to get further in the tutorial.

The first part of the manual comprises texts introducing users to the specified issue's plot. The functionality is explained in more detail in the videos at each level. At most levels, practical tasks also verify and practice knowledge from a specific unit. The last level serves to review the acquired knowledge. Its content is a minigame (**Figure 13**) that contains intentional errors. To complete the level, the user must find and correct these errors, while the upper right corner of the screen shows how many errors he has already fixed.

The length of the instructions is interesting. While some students spent 10 h with it, others solved it in 1 h. Therefore, in the case of the Godot engine, we have followed the classic instructions.

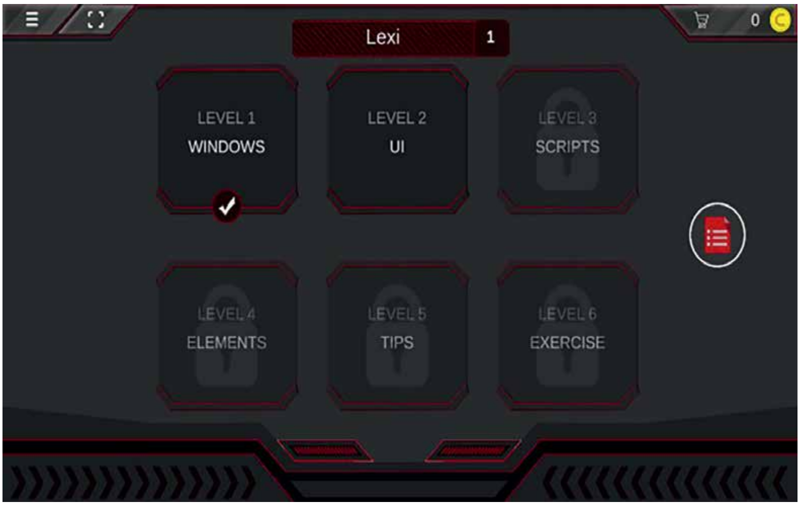


Figure 12.
Display screens in the manual.

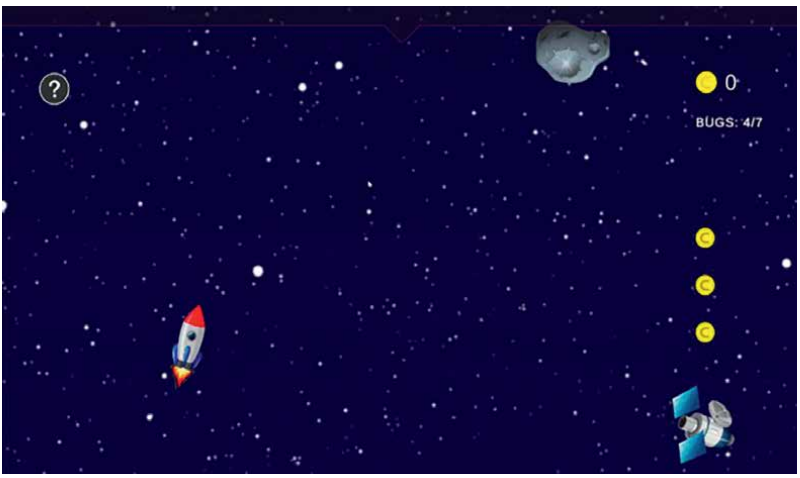


Figure 13.
The example of the minigame in the last level.

4.3 Step-by-step tutorial for Godot engine

In the next part, we will describe a detailed guide to familiarize students with the basics of game development in the Godot engine. The goal was to provide a guide detailed enough to easily guide students through the essential elements of the game through practice. We wrote it so that it was obvious what was expected from the students and that they needed a minimum of assistance from the teacher. At the same time, we have included enough space for students' creativity in the instructions.

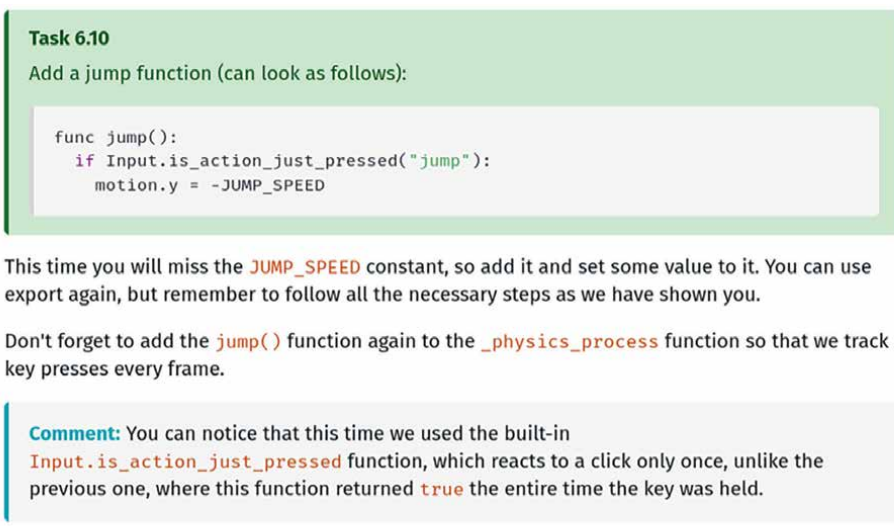
Students have mastered a playable simple 2D platform game by fully completing the tutorial. They have a player with animations, a level composed of tiles, a background with a parallax effect, a functional collision system, an interactive object, a moving camera, or gravity. The instructions[29] are divided into the fundamental part, where the students are fully guided, and the supplementary part, which consists of additional tasks. The idea was to explain and let the students practically try out all the basic functionalities of the Godot engine so that they could playfully continue their own in the additional part. And if they start additional tasks, the number of collected objects and the number of lives that can be lost or gained in diverse ways are recorded in the user interface (UI), and there is also a way to win the game (**Figure 14**).

Although the instructions are long, the students were fine with it. Compared to Unity's interactive tutorial, the solution time has stabilized at 3–4 h. That is why we have returned to the standard tutorial[30] for the Unreal engine.

4.4 Step-by-step tutorial for unreal engine

The difference between a step-by-step and an interactive tutorial is that the individual steps are not described inside the engine environment but in a separate tutorial. The difference between the step-by-step instructions for the Godot engine and UE was that while with Godot, we got by with written instructions with a minimum of figures, with Unreal, it was easier to use figures due to the structure of the environment (**Figure 15**).

The effect was as expected; the students spent a similar amount of time, 3–4 h, solving the tasks.



Task 6.10

Add a jump function (can look as follows):

```
func jump():
  if Input.is_action_just_pressed("jump"):
    motion.y = -JUMP_SPEED
```

This time you will miss the `JUMP_SPEED` constant, so add it and set some value to it. You can use export again, but remember to follow all the necessary steps as we have shown you.

Don't forget to add the `jump()` function again to the `_physics_process` function so that we track key presses every frame.

Comment: You can notice that this time we used the built-in `Input.is_action_just_pressed` function, which reacts to a click only once, unlike the previous one, where this function returned `true` the entire time the key was held.

Figure 14.
The task example in the Godot engine tutorial.

[29] Godot's step-by-step tutorial: [cit. 2023-06]. URL: https://kurzy.kpi.fei.tuke.sk/gamedev/en/book/godot.html

[30] Unreal's step-by-step tutorial: [cit. 2023-06]. URL: https://kurzy.kpi.fei.tuke.sk/gamedev/en/ book/unreal.html

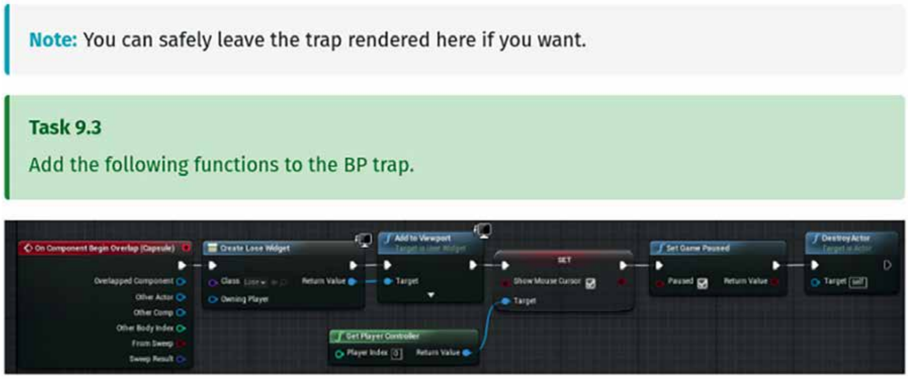


Figure 15.
The task example (trap functionality) in the Unreal Engine (UE) tutorial.

4.5 Impact on learning outcomes

We have mentioned that game engines offer various features suitable for teaching styles and learning preferences. Now, we will discuss how these differences in features impact learning outcomes. The effects of these differences depend on several factors, like individual learning styles, project nature, or instructional design. For instance, game engines such as Unity, with their user-friendly interface and intuitive design, may reduce cognitive load for beginners, enabling them to grasp fundamental concepts more quickly. This reduced cognitive load can provide an environment where learners can experiment freely and learn by doing, leading to increased engagement and improved understanding of core game development concepts.

On the other hand, more complex game engines, like Unreal Engine, offer advanced functionalities that can provide opportunities for deep learning and mastery for more advanced learners. The complexity of these engines could promote a deeper understanding of the intricate mechanics of game development, enhancing problem-solving and critical-thinking skills.

Table 2 showcases the impact of the three different tutorial approaches on various learning outcomes, highlighting their unique strengths and features. For instance, Unity's interactive tutorial stands out for its emphasis on interactive exercises and gamification, while the step-by-step tutorials for Godot and Unreal Engine encourage independent learning and reinforcement through repetition. Visual learning support is a unique strength of the Unreal Engine.

5. Summary and annual game jam experience

Since games intersect many disciplines, including art, sound, storytelling, or design, the game engines allow the introduction of these elements next to the more technical aspects of computer science. The tutorial's diversity demonstrates game engines' adaptability to different learning styles and underscores their value as practical teaching tools. Moreover, Unity and Unreal engines are known for their comprehensive VR support, offering various built-in features.

Learning outcomes	Interactive tutorial in Unity	Step-by-step tutorial for Godot	Step-by-step tutorial for Unreal Engine
Engine understanding	X	X	X
Active learning	X	X	X
Self-paced learning	X	X	X
Problem-solving skills	X	X	X
Feedback and reflection	X	X	X
Interactive exercises	X		
Motivation through gamification	X		
Reinforcement through repetition		X	X
Independent learning		X	X
Visual learning support			X

Table 2.
Tutorials' impact on learning outcomes.

A highlight of our exploration is the hands-on approach to learning, represented by interactive and step-by-step tutorials in game engines. Each tutorial approach offers unique insights. The Unity tutorial adopted an interactive learning approach, reinforcing theoretical concepts with practical application and fostering a dynamic understanding of game development in Unity. Such a tutorial suits learners who prefer active learning and are comfortable with some degree of self-direction. Given its interactive nature, the time spent on tasks can vary among students, ranging from 1 to 10 h. This variability reflects the tutorial's exploratory nature and the learners' potential struggles within the engine environment.

On the other hand, the Godot tutorial, a step-by-step guide, allowed readers to build their understanding progressively, each step building upon the last to create a complete picture of game development in Godot. Such a tutorial is ideal for a beginner or those who prefer a structured approach to learning. With clear instructions, students spend a stable amount of time on tasks, approx. 3–4 h. This stability indicates the straightforward, well-defined nature of the tutorial and students' habit of working with structured documentation. In our case, structured materials are a natural part of institutional courses.

Compared with the Godot tutorial, the Unreal Engine tutorial focused on a step-by-step process combined with visual aids. However, instead of programming the scripts, we used blueprints. Such figure-based learning benefits visual learners, enabling a clear understanding of complex processes involved in game development using the engine. Like the Godot tutorial, the time commitment for tasks is stable, with students spending approx. 3–4 h. The consistency in time spent suggests that learners can expect a steady pace of progress.

From 2018 to 2020, our curriculum focused more broadly on game design and programming technologies. However, reflecting increased student interest and the industry's evolving needs, we began integrating specific game engines into our curriculum in 2021, as shown in **Table 3**. By 2023, we had successfully incorporated the Unity, Godot, and Unreal Engine into our syllabus.

Table 4 shows our students' changes in technology preferences during our annual game jam events, alongside participation numbers. Unity was the dominant choice

Year	Unity	Godot	Unreal Engine
2018–2020	—	—	—
2021	X	—	—
2022	X	X	—
2023	X	X	X

Table 3.
An overview of engine coverage over the period.

Year	Participants	Teams	Unity (%)	Godot (%)	Unreal Engine (%)	Others (%)
2018	76	19	11	0	0	89
2019	69	18	40	0	0	60
2020	68	17	88	0	0	12
2021	65	17	82	18	0	0
2022	72	18	39	61	0	0
2023	168	42	34	42	10	14

Table 4.
An overview of technology preferences and game jam participation over the period.

Engine		None (%)	Low (%)	Moderate (%)	High (%)	Very high (%)
Unity	Before	73	11	6	4	6
	After	0	15	54	25	6
Godot	Before	100	0	0	0	0
	After	1	21	53	22	3
Unreal Engine	Before	86	8	6	0	0
	After	1	32	44	20	3

Table 5.
Overview of skills according to students' questionnaire.

in 2020, but with the introduction of Godot into our curriculum in 2021, we saw a shift in preference. By 2023[31], we saw a more balanced distribution among the three engines, demonstrating that our students acquired diverse skills. Of note, the total number of participants and teams substantially increased in 2023, highlighting the growing interest in our game design and development courses.

As **Table 5** illustrates, incorporating game engines into our curriculum has marked improvements in our students' experience with the engines. Based on a questionnaire, most students reported having no or low skill levels at the beginning of the semester. However, by the end of the semester, a significant shift toward moderate and high skill levels is observable. The data suggest our curriculum's focus on hands-on

31 Games of 2023's Game Jam: [cit. 2023-06]. URL: https://itch.io/jam/game-jam-kosice-2023

learning with various game engines equips our students with applicable skills in game development, preparing them for diverse opportunities in the industry.

These results indicate the effectiveness of different game engines in teaching game development, each offering unique strengths to meet diverse learner needs and project requirements.

6. Conclusion

In this chapter, we have explored the role of game engines in game development and learning. Through numerous examples, we have illustrated how these engines employ their powerful capabilities to craft immersive digital experiences. Importantly, we have highlighted how game engines can extend beyond creation into education as practical game design tools.

In our department, we focus on education in software engineering. So, in the design and development of games, the implementation side, including the design process (in this sense, also the use of game engines), is vital to us. We are looking for ways and activities to teach students to transfer formalisms and critical aspects of Game Theory into Computer science practice. It is not only a transfer into source code but also a design or feedback.

The choice of an appropriate game engine significantly impacts a game's performance, capabilities, and overall development process. When considering the educational impact, tutorials in these engines give students a practical understanding of game design and development, equipping them with skills highly demanded in the industry. Our insights from exploring interactive and step-by-step tutorials in Unity, Godot, and Unreal Engine further emphasized this point. The variability in the time spent on tasks across these tutorials reflected their adaptability to different learning styles.

As demonstrated in our curriculum evolution and students' improved experience in game engine technologies, our teaching approach empowers students with the critical skills needed. Furthermore, it fosters a dynamic learning environment full of motivation and creativity. As we adapt to technological advances and align with industry trends, we envision a responsive, anticipative, and nurturing future of game design education.

Moreover, our annual game jam event revealed these engines' real-world relevance and application in a learning context. The engine preferences of our students underscored the need for a diverse toolkit in game development education. Integrating game engines into the curriculum can provide more engaging and immersive learning experiences, revolutionizing the way we teach game design and computer science.

Understanding and exploring game engines is crucial for anyone interested in game development. It offers a blend of creativity and technical skills, fostering a comprehensive approach to learning. We anticipate further research and exploration in this exciting field, keeping up with the continuous evolution of game development technologies.

Acknowledgements

This work has been supported by Slovak KEGA Agency under Grant No. 048TUKE-4/2022: "Collaborative virtual reality technologies in the educational process," under Grant No. 002TUKE-4/2021: "Implementation of Modern Methods and Education Forms in the Area of Cybersecurity towards Requirements of Labour Market" and the APVV Grant No. APVV-21-0105 "Trustworthy human–robot and therapist–patient interaction in virtual reality."

Author details

Branislav Sobota* and Emília Pietriková
Technical University of Košice, Slovakia

*Address all correspondence to: branislav.sobota@tuke.sk

References

[1] Adams E. Fundamentals of Game Design. 3rd ed. New Riders; 2013. ISBN 978-0321929679

[2] Schnell J. The Art of Game Design: A Book of Lenses. 3rd ed. CRC Press; 2019 ISBN 978-1138632059

[3] Fullerton T. Game Design Workshop: A Playcentric Approach to Creating Innovative Games. 4th ed. CRC Press; 2018. ISBN 978-1482217162

[4] G. Junker, Pro OGRE 3D Programming. 1st ed. Berkeley, CA (USA): Apress; 2006. p. 312. ISBN 1590597109

[5] F. Kerge, OGRE 3D 1.7 Beginner's Guide. Packt Publishing; 2010. p. 300 ISBN 1849512485

[6] Juarez A, Schonenberg W, Bartneck C. Implementing a low-cost CAVE system using the CryEngine2. Entertainment Computing. 2010;**1**:157-164

[7] Koscianski A. Game Programming with Irrlicht. CreateSpace Independent Publishing Platform; 2011. p. 194. ISBN 1460905148

[8] Stein J, Kyaw AS. Irrlicht 1.7 Realtime 3D Engine Beginner's Guide. Packt Publishing; 2011. p. 272 ISBN 1849513988

[9] Rosipal R, Korečko Š, Rošťáková Z, Porubcová N, Vanko M, Sobota B. Towards an ecologically valid Symbiosis of BCI and head-mounted VR displays. In: Proceedings of 16th IEEE International Scientific Conference on Informatics; Poprad—High Tatras. November 23-25; Danvers, MA (USA), IEEE; 2022. pp. 251-256. ISBN 979-8-3503-1033-7

[10] Manzur A, Marques G. Godot Engine Game Development in 24 Hours, Sams Teach Yourself: The Official Guide to Godot 3.0. Pearson Education. 2018. ISBN 9780134835211

[11] Johnson B. Using OGRE as a Means of Teaching C plus plus Programming. In: ACM-SE '11: Proceedings of the 49th Annual Southeast Regional Conference, Article No. 2. ACM, New York (USA); 2011. p. 1

[12] Shreiner D, Angel E. There and Back Again: Graphics Education with a View Towards the Future. In: SIGGRAPH '22: ACM SIGGRAPH 2022 Educator's Forum, Article No. 17. New York (USA): ACM; 2022. pp. 1-2

[13] Karaoglu EO, Tükel D, Arthaya B. VR Based Visualization of Robotic Workcells using Cryengine. In: International Conference on Mechatronics, Robotics and Systems Engineering (MoRSE). Danvers, MA (USA): IEEE; 2019. pp. 118-121

[14] Fachada N, Códices N. Top-down design of a CS curriculum for a computer games BA. In: ITiCSE '20: Proceedings of the 2020 ACM Conference on Innovation and Technology in Computer Science Education. New York (USA): ACM; 2020. pp. 300-306

[15] Gestwicki P. Unreal engine 4 for computer scientists. Journal of Computing Sciences in Colleges. New York (USA): ACM; 2019;**35**(5):109-110

[16] Gestwicki P. Godot engine and checklist-based specifications: Revising a game programming class for asynchronous online teaching. Journal of Computing Sciences in Colleges. New York (USA): ACM; 2021;**37**(4):30-40

[17] Hodges CB, Akcaoglu M, Allen A, Dogan S. Teacher self-efficacy during professional development for game design and unity. In: SIGCSE 2022: Proceedings of the 53rd ACM Technical Symposium on Computer Science Education V.2. New York (USA): ACM; 2022. p. 1144

Chapter 6

Elementals, A Chemistry Inclusive Serious Game

Paula Escudeiro and Márcia Campos Gouveia

Abstract

Promoting equity and social and educational inclusion of disabled people is a priority of European Education. Disabled students often feel challenged to communicate or access information. However, with the emergence of new technologies, disabled students are offered more chances to have the same learning opportunities as others. Our current work contributes to the field of Educational Serious Game. Elementals, a chemistry-inclusive serious game, aims to assist all students in the learning process. It was designed to be inclusive for players with visual, hearing, and mobility impairments. Its purpose is to help students with chemistry and to be a complementary solution to common learning methods to enhance education. In our chapter, we will provide information about this developed game that educates, informs, entertains, and motivates chemistry students in their learning experience through a card game in which they learn the Table of Periodic Elements and use them to form molecules.

Keywords: educational serious games, inclusion, accessibility, chemistry, gamification

1. Introduction

Nowadays, education continues to be a challenge for deaf students, as they try to advance in their academic activities and future professional careers [1]. Back in 1965, the learning process for individuals with disabilities was challenging. However, the learning process for deaf individuals remains inherently demanding, mainly due to communication problems, emphasizing the critical need for new approaches that can help overcome barriers to teaching and learning [2]. The aim of new approaches is not only to enhance the classroom experience for these students but also to facilitate their ongoing educational journey in self-learning environments. In this context, the development of inclusive and educational tools is of utmost importance.

One such tool is “Elementals,” an innovative serious game that has been developed to address the numerous challenges faced by deaf students and promote their inclusion in educational settings. Recognizing the difficulties encountered by students in learning chemistry, Elementals offers a complementary solution to traditional teaching methods, mitigating the reluctance often associated with the intensive reading of textbooks. By employing engaging gameplay elements and a collectible card system, Elementals actively educates and informs students, fostering a deep understanding of such complex subject matter, through entertainment.

Besides being an educational tool that helps students to learn the Table of Periodic Elements, the game also serves as a catalyst for raising awareness about the learning challenges experienced by students with disabilities [1]. With an inclusive design approach, the game addresses the needs of individuals with color blindness, mobility impairments, and hearing impairments, ensuring accessibility for all students. This is made possible through the implementation of technologies such as augmented reality and georeferencing [3]. Georeferencing, as highlighted in Hackeloeer's publication from 2014, enables precise association of geographic coordinates with spatial data. By leveraging georeferencing techniques, the game seamlessly integrates virtual elements into real-world environments, creating an immersive and inclusive gaming experience.

By addressing the challenges specific to chemistry education, the game opens doors to enhanced educational experiences and fosters equal opportunities for all students. Through ongoing research and innovation, the researchers are committed to empowering students with diverse abilities and paving the way for a more inclusive education system [1].

Following this line of thought, the chapter is organized into different sections and subsections. Firstly, the researchers will provide a summarized background/framework of their work, followed by a methodology section, where the concept and objectives of this work are outlined. In addition to these, the chapter also includes a Game Development section that focuses on the detailed process of developing Elementals. The researchers discuss the various stages of game development, including design, programming, and implementation, highlighting the key features and mechanics incorporated into the game to ensure its effectiveness as an educational tool.

Finally, the researchers included a Game Evaluation and Quality Scenario Control. In this section, the researchers delve into the evaluation and quality control aspects of the game. They discuss the evaluation methodology employed to assess the effectiveness and impact of the game. The researchers explain the use of the Quality Evaluation Framework (QEF) as a tool to measure the educational value, accessibility, and inclusivity of the game. The section presents the findings of the evaluation process, including feedback from users and the overall performance of the game in meeting its intended objectives.

By including these sections, the chapter provides a comprehensive overview of the framework of the game development, the game development process itself, and the evaluation and quality control measures undertaken to ensure the effectiveness and inclusivity of Elementals.

2. Framework: the use of serious games in education

Access to equal opportunities and social inclusion for disabled individuals is a major concern in modern society and a key issue within European education. In response to this, significant efforts are being made to enhance education through the implementation of educational and social inclusion initiatives. The concept of inclusion, in this context, aims on ensuring that all students, regardless of their abilities, can actively participate and thrive in mainstream educational settings. Achieving this goal requires comprehensive changes in pedagogies within the education system [4].

Addressing these challenges requires the incorporation of new technologies to drive positive transformations in educational settings. These technologies can introduce valuable opportunities for social inclusion, as part of a holistic approach. It becomes crucial not only to improve the education system and the overall educational settings but also to promote the utilization of new technologies to assist students in their studies and self-learning environments.

In this regard, the development of Educational Serious Games has emerged as a highly promising and innovative approach to enhancing education and inclusion. These serious games are specifically designed with the primary goal of education in mind, setting them apart from conventional entertainment-focused games. They are carefully crafted to serve an educational purpose, aiming to develop knowledge, skills, or specific learning outcomes. Throughout the last two decades, there has been a significant emphasis on developing educational games that not only serve the primary purpose of enhancing knowledge but also aim to be engaging and entertaining. These games have been designed with the intention of captivating players and fostering motivation, thereby promoting active participation [5–7].

An essential aspect of serious games is their ability to align with their educational purpose, guiding players toward meaningful learning experiences. They offer distinct and effective mechanisms for learning and assessment, which set them apart from traditional instructional methods [2].

The relationship between game development, computer science, and disabled people's education is a powerful and transformative one. Through the convergence of these fields, innovative solutions can be created to address the unique challenges faced by individuals with disabilities, ultimately fostering inclusivity and equal access to education. Let us explore this relationship in more detail:

1. The Role of Game Development: Game development plays a pivotal role in shaping educational experiences, especially for students with disabilities. By leveraging interactive and immersive gameplay, game developers can create engaging environments that capture the attention and interest of learners. Games provide a safe and enjoyable space for individuals to explore complex concepts, experiment with different approaches, and reinforce their understanding through active participation. The integration of educational content within games can transform traditional learning into a dynamic and accessible experience.

2. The Intersection with Computer Science: Computer science serves as the backbone of game development, providing the technical foundation for creating interactive experiences. In the context of education and disability empowerment, computer science brings forth powerful tools and technologies that can be harnessed to enhance accessibility and inclusivity in games. From adaptive control systems and assistive technologies to data analysis and machine learning algorithms, computer science empowers game developers to create tailored solutions that accommodate diverse learning needs and overcome barriers for disabled individuals.

3. Empowering Disabled People through Education: Education is a fundamental right, and it is crucial to ensure that individuals with disabilities have equal access to quality educational opportunities. Game development, driven by computer science advancements, can significantly contribute to empowering disabled people in education. By incorporating accessibility features and adaptive technologies, games can provide tailored experiences that cater to specific disabilities, allowing individuals to actively engage and participate in the learning process.

This inclusion not only facilitates knowledge acquisition but also fosters confidence, self-expression, and social interaction.

The relationship between game development, computer science, and disabled people's education goes beyond just creating accessible games. It opens doors to new possibilities, empowering individuals to overcome limitations, develop critical skills, and thrive in various educational domains. Through the use of adaptive technologies, such as brain-computer interfaces or augmented reality, games can provide unique learning experiences that bridge gaps and promote independence for disabled learners. Moreover, the collaboration between game developers, computer scientists, and educators can lead to continuous innovation, driving the evolution of educational approaches that prioritize inclusivity and personalized learning.

By embracing educational serious games and leveraging their potential, we can advance educational and social inclusion by providing interactive and effective learning experiences for all students. These games serve as a bridge between education and entertainment, offering a powerful tool for engagement and knowledge acquisition [6, 7]. Through their thoughtful design and purposeful implementation, serious games contribute to creating an inclusive and enriching educational environment for both disabled and non-disabled students alike.

3. Methodology

The Elementals game is designed to enhance the learning process for chemistry students, providing them with valuable support that complements common teaching and learning methods. Its overarching purpose is to be educational and inclusive. With a strong focus on accessibility and social inclusion, the Elementals game strives to ensure that students and players with disabilities can actively engage with the content and participate in the learning experience. The Elementals game not only enriches the educational journey of junior and high-school chemistry students but also emphasizes the importance of equal opportunities for all learners.

Through its carefully crafted methodology and project outcomes, the Elementals game makes a significant contribution to the advancement of inclusiveness in education. Integrating accessible features in response to the needs of different learners creates an environment that fosters engagement, comprehension, and knowledge acquisition.

3.1 Concept

The Elementals project sets out to create an immersive and educational serious game that revolves around the concept of collecting cards. The game will offer multiplayer support, enabling players to engage and compete with one another, and it will leverage cutting-edge technologies such as augmented reality and geolocation.

The success of the game will be measured by its impact on academic performance, as it aims to improve learning outcomes through the application of gamification principles and a less intimidating approach to the study of chemistry. The mechanisms and features developed for the game will be carefully crafted to ensure inclusivity, allowing students with disabilities to effortlessly engage with the game.

At the core of the game's objective is the collection of molecule cards, with players striving to increase their ranking by accumulating a diverse range of cards. To obtain these molecule cards, players must first gather atom cards that are dispersed across real-world map locations. Leveraging their chemistry knowledge and skills, players strategically select and combine atom cards to construct unique and valuable molecule cards, which are then added to their ever-expanding collection.

To assess the presence of any disabilities, the game begins with a puzzle-based evaluation. This assessment includes challenges designed to determine color blindness, featuring squares of varying sizes and colors, as well as options for players to select their preferred communication mode, such as written or sign language, to identify any hearing impairments.

The game takes players on an exploration journey starting from their current geographic location, searching for atom cards scattered throughout the real-world map [3]. The level of engagement and success in finding hidden atoms depends on the players' determination and their familiarity with gameplay mechanics. Once close to an atom card, players can activate the augmented reality feature to view the atom directly within their surroundings and add it to their card collection. Additionally, players can strategically combine sets of atom cards to create unique and powerful molecule cards, earning valuable score points that contribute to their leaderboard ranking. This leaderboard serves as a representation of each player's progress and learning achievements.

To provide ongoing motivation and rewards, the game presents daily objectives to players based on their ranking. By completing these objectives, players unlock rewards and advance their user level. As players ascend to higher levels, they gain access to crafting more complex molecules that were previously beyond their reach. This progressive system not only enhances the learning process but also encourages continuous skill development and deepens their understanding of chemistry.

The Elementals project strives to revolutionize the educational gaming landscape by offering an inclusive and captivating experience that seamlessly integrates learning and entertainment.

3.2 Objectives

Elementals is a groundbreaking game that has been designed to revolutionize education and ensure that every student has access to equal opportunities. It seeks to achieve two main objectives: developing an educational component focused on the field of chemistry and fostering inclusivity for players with disabilities, including color blindness, and mobility and hearing impairments. By incorporating innovative features and adaptations, Elementals empowers individuals with disabilities to fully participate in the game. Thus, the game aims to incorporate the following specific objectives:

1. Educational component: The game offers a unique approach to learning chemistry by complementing traditional schoolbooks with a card-collecting system. This engaging gameplay mechanic encourages students to actively explore and acquire knowledge about chemical elements and compounds. By collecting and organizing cards, players gain a deeper understanding of the subject matter and develop essential skills.

2. Inclusivity for players with disabilities: One of the key aspects of Elementals is its commitment to allowing players to configure the game according to their specific needs. Importantly, this configuration is entirely optional and not compulsory,

ensuring that players can choose to customize the game to their preferences. This empowers individuals to create an experience that aligns with their unique requirements and learning styles.

3. Inclusion of color-blind players: To promote the inclusion of color-blind players, Elementals incorporates feedback mechanisms that go beyond traditional color-based cues. Instead, the game utilizes well-defined and easily understandable symbols that provide clear and unambiguous information. By relying on these symbols, color-blind players can fully comprehend and interact with the game's elements without facing any disadvantages.

4. Inclusion of deaf players: Elementals embraces the inclusion of deaf players by providing feedback that caters to their specific needs. This involves the use of sign language to express simple messages, which are then translated into sequences of sign language characters within the game. By incorporating sign language, the game ensures effective communication and enables deaf players to fully engage with the gameplay experience.

5. Inclusion of players with mobility impairments: Elementals is dedicated to supporting the inclusion of players with mobility impairments. It achieves this by integrating an EPOC device, which enables alternative input methods. These methods may include mimicking actions, utilizing shortcuts, or employing push-buttons to interact with the game. By providing full user accessibility, the game ensures that players with mobility limitations can navigate through the game world and participate in all aspects of gameplay.

Overall, Elementals represents a groundbreaking approach to education and inclusivity. By focusing on chemistry and incorporating features specifically tailored to individuals with disabilities, the game breaks down barriers and promotes equal opportunities for all players. Elementals empowers students to actively learn, raises awareness of learning challenges, and fosters a sense of inclusion and accessibility within the gaming community.

4. Game development

The game architecture of Elementals comprises six primary components, each playing a crucial role in delivering a cohesive and immersive gameplay experience. These components work together seamlessly, leveraging cutting-edge technologies to achieve the game's objectives and provide an engaging learning environment for players.

4.1 Game architecture

The Elementals game encompasses six primary components (**Figure 1**) that perform together to create a cohesive and immersive experience. These components are carefully designed to fulfill specific functionalities and contribute to the overall game objectives. These components are as follows:

1. Mobile Application Component: The Mobile Application Component serves as the user-facing interface of the game. It enables users to interact with the game and

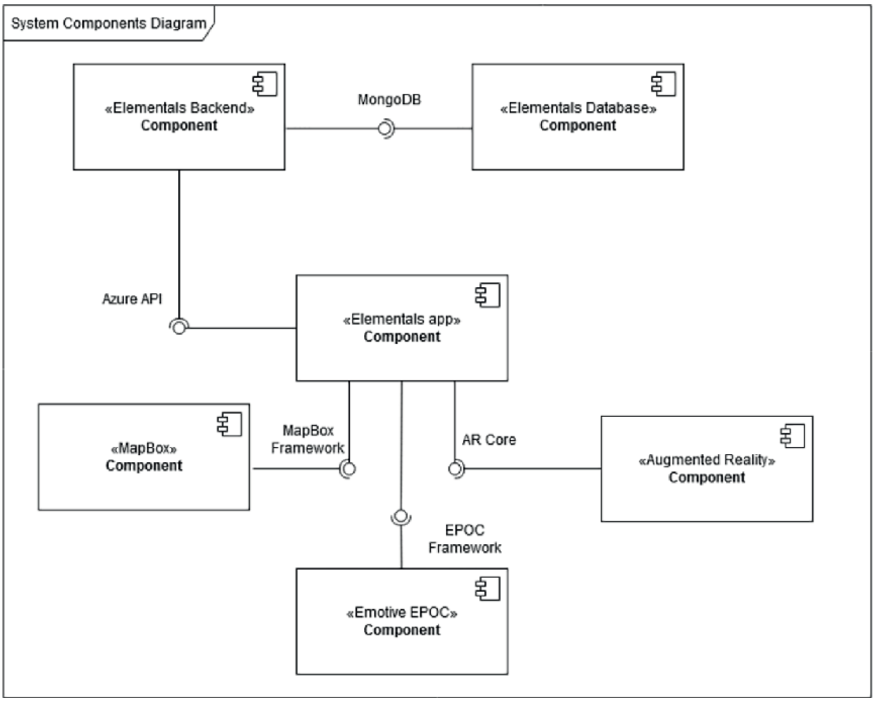


Figure 1.
System component diagram.

access various features and functionalities. This component establishes communication with the backend component to exchange data and trigger actions. The backend component provides a set of services that the mobile application can consume.

2. Backend Component: The Backend Component consists of a Node server hosted on the Azure platform. It acts as an intermediary between the Mobile Application component and the other game components. This component offers a well-defined interface, exposing a range of services that the mobile application can utilize. Through this backend component, the mobile application can access and manipulate data, perform calculations, and interact with the wider system.

3. Database Component: The Database Component, implemented using MongoDB, plays a vital role in storing and retrieving essential information required by the game. The Backend Component communicates with the Database Component to fetch the necessary data. MongoDB, a flexible and scalable NoSQL database, provides efficient storage and retrieval mechanisms, ensuring the game's smooth functioning.

4. MapBox Component: The MapBox Component configures and generates the world map, incorporating real coordinates and serving as a foundational layer for realism and immersion. This component enhances the gameplay experience by providing a dynamic and interactive environment where players can explore, search for atoms, and encounter other players. The MapBox component contributes to the game's spatial awareness and location-based features.

5. Emotive EPOC (Brain-Computer Interfaces) Component: The Emotive EPOC Component serves as an interface specifically designed to cater to individuals who may have physical movement limitations. It allows users to interact with the application using inclusive inputs based on brain-computer interfaces. This component ensures that the game remains accessible to a wider range of users, promoting inclusivity and equal participation.

6. Augmented Reality Framework Component: The Augmented Reality (AR) Framework is a crucial component that brings an immersive experience to the game. It enables players to collect atoms in a visually captivating and interactive manner. By leveraging AR technology, this component overlays virtual elements onto the real world, enhancing the gameplay and fostering a deeper sense of engagement.

Each of these components within the Elementals game serves a distinct purpose and contributes to its overall functionality and user experience. The seamless integration and collaboration among these components create a cohesive and compelling environment for players to learn and enjoy the game.

4.2 Special technical requirement

For this game, the researchers applied a range of technologies and tools to meet the technical requirements. These include:

1. Unity 3D: The researchers utilized the Unity 3D engine as the foundation for developing the game. Unity 3D offers a comprehensive set of tools and features for creating interactive and immersive experiences.

2. Visual Studio Code: This code editor was employed by the researchers to write and edit the game's source code. Visual Studio Code provides a lightweight and versatile environment with support for various programming languages.

3. Visual Studio Community Edition 2019: The researchers used this integrated development environment (IDE) for building and debugging the game. Visual Studio Community Edition offers a rich set of features for software development.

4. Map Box—Georeferencing Services: Map Box's georeferencing services were incorporated by the researchers to enable location-based functionalities within the game. These services provide access to accurate and up-to-date mapping data.

5. Photon: Photon was used for implementing multiplayer networking capabilities in the game. This framework facilitates real-time communication and synchronization between multiple users.

6. Blender for 3D modeling: The researchers employed Blender, a powerful open-source software, for creating and editing 3D models used in the game. Blender offers a wide range of tools and functionalities for modeling, texturing, and animating objects.

7. REST Services for server communication: REST (Representational State Transfer) services were utilized by the researchers for server communication. RESTful

Application Programming Interfaces (APIs) enable efficient and standardized communication between the game's client-side and server-side components.

8. Server and Database: The researchers set up a server infrastructure to handle data storage and processing requirements. Additionally, a database system was employed to store and manage game-related data efficiently.

9. Microsoft Paint 3D: Microsoft Paint 3D was used by the researchers as a supplementary tool for creating or modifying 3D assets. This software allows for basic 3D modeling and texture editing.
10. InkScape: InkScape, a vector graphics editor, was employed by the researchers for creating and manipulating scalable 2D graphics. It provides a comprehensive set of tools for designing and editing vector-based artwork.
11. EPOC Unity Plugin: The EPOC Unity Plugin was integrated into the game by the researchers to interface with the EPOC Emotiv headset. This plugin allows for capturing and processing data from the EPOC Emotiv headset, enabling brain-computer interface functionality.
12. Emotiv App—EPOC Emulator: The Emotiv App was employed as an emulator for the EPOC Emotiv headset. This software simulates the functionality of the physical headset, enabling development and testing without the actual hardware.
13. Vuforia—Augmented Reality SDK: Vuforia, an Augmented Reality Software Development Kit (SDK), was selected to implement augmented reality features within the game. It provides tools and APIs for marker-based and markerless AR experiences.

By incorporating these technologies and tools, the researchers ensured that the game was equipped with the necessary features and functionalities to deliver a dynamic, inclusive, accessible, and engaging user experience.

4.3 Game engine

The researchers selected Unity 3D as the game engine for this game, taking into consideration several key factors. Firstly, Unity 3D was chosen because it is designed to be multiplatform, meaning that the game can be developed and deployed on various operating systems and devices. This multiplatform capability ensures a wider reach and accessibility for players.

Additionally, Unity 3D offers an integrated development environment (IDE), which provides a seamless and efficient workflow for game development. The IDE includes a comprehensive set of tools and features that streamline the development process, allowing the researchers to focus on creating engaging gameplay experiences. Furthermore, Unity 3D comes with an extensive asset library, providing a rich collection of pre-built game objects, resources, and visual effects. This asset library significantly enhances the development speed and efficiency, as it offers a wide variety of ready-to-use components.

Unity 3D has gained immense popularity within the gaming community in recent years. It has become one of the most widely adopted game engines due to its versatility and ease of use. Developers appreciate its flexibility and adaptability, enabling them to create games for different genres and platforms with relative ease. Moreover, Unity 3D has an active and vibrant community that actively contributes to its growth and provides valuable resources, tutorials, and support.

One of the significant advantages of Unity 3D is that it is an open-source engine, which means that it is freely available for developers to use and modify. This open-source nature encourages collaboration, innovation, and the sharing of knowledge among developers. Additionally, the free-to-use model makes Unity 3D accessible to developers of all backgrounds, irrespective of their budgetary constraints.

Considering these factors, the researchers employed Unity 3D as the game engine for this game, leveraging its multiplatform capabilities, integrated development environment, extensive asset library, widespread popularity, versatility, user-friendly interface, and open-source nature to meet the requirements of the game.

4.4 Rendering

Blender's 3D modeling engine played a crucial role in the development and design of various components within the game. Specifically, the researchers utilized Blender to create and design the Atom models, Molecule models, Cards, and other objects featured in the game.

Blender, a powerful and versatile software, provided the researchers with a comprehensive set of tools and functionalities to bring these objects to life in a three-dimensional space. The Atom models, representing the building blocks of matter, were meticulously crafted using Blender's modeling tools, allowing for precise detailing and accurate representation.

Similarly, the Molecule models, which showcase the complex structures and interactions of chemical compounds, were developed using Blender's advanced features. The researchers leveraged Blender's capabilities to accurately depict the intricate bonds, atoms, and molecular arrangements, ensuring a visually compelling and scientifically accurate representation.

In addition to the Atom and Molecule models, Blender was instrumental in designing other objects essential to the game's gameplay and visual esthetics. Cards, for instance, were crafted using Blender's 3D modeling engine to create distinct and visually appealing playing cards with intricate designs, textures, and animations.

Blender's extensive capabilities in 3D modeling, texturing, and animation enabled the researchers to unleash their creativity and bring their vision to life. The software's intuitive interface and the vast array of tools facilitated the creation of high-quality and realistic objects, enhancing the overall immersion and visual fidelity of the game.

By leveraging Blender's powerful 3D modeling engine, the researchers were able to develop visually captivating and engaging components such as Atom models, Molecule models, Cards, and various other objects.

4.5 Gameplay objects

The gameplay objects in this game were divided into two distinct major components, namely menus for navigation and a game canvas where the actual card collection levels are played. The separation of menus and the game canvas was carefully implemented, following the guidelines provided in Unity's documentation. This approach aimed to create a structured pattern that efficiently isolates different layers of the game, facilitating organization and maintenance.

To achieve this separation, the development process involved creating the menus and game canvas' as separate entities. The menus component focused on designing and implementing the user interface elements responsible for navigation, such as main menus, options menus, and level selection screens. These menus were developed following the recommended Unity documentation guidelines, ensuring consistency and usability.

On the other hand, the game canvas component encompassed the actual gameplay levels where players engage in card collection. The game canvas was implemented as a distinct area within the game, dedicated to providing an immersive and interactive experience. The development of the game canvas followed Unity's recommended practices, taking advantage of the available tools and features to create engaging and challenging levels.

To further streamline the development process, the game's development components were organized into a structure. This structure encompassed the following elements:

1. Game actions: This component involved defining the various actions and interactions that players could perform within the game. Examples include card selection, shuffling, and scoring mechanisms. The game actions component served as the foundation for implementing the gameplay mechanics.

2. Game action scripts: These scripts were responsible for implementing the specific functionalities associated with each game action. By utilizing scripts, the researchers could define the behavior and logic behind the gameplay mechanics, ensuring smooth and accurate execution.

3. Graphic Resources: This component encompassed the visual assets used in the game, including textures, sprites, animations, and other graphical elements. The graphic resources component played a vital role in creating visually appealing and engaging gameplay experiences.

4. Graphic Assets: This component involved the creation and management of graphic assets, such as card designs, backgrounds, and visual effects. The researchers carefully crafted these assets to enhance the esthetics of the game and provide an immersive visual experience.

5. Game Engine controllers: The game engine controllers served as the bridge between the various development components and the underlying Unity game engine. These controllers were responsible for managing the flow of the game, handling input events, managing game states, and ensuring proper integration between different elements of the game.

By organizing the development components into this structure, the researchers ensured a systematic and efficient approach to the game's implementation.

4.6 Game data storage and tracking

The gameplay data, specifically the scores and ranking levels of each player, were stored and managed in a dedicated Servers Database deployed on the Microsoft Azure platform. This cloud-based database solution provided a reliable and scalable infrastructure for securely storing and retrieving game-related information.

To facilitate seamless communication between the game user and the server database, a REST (Representational State Transfer) API was developed. This API utilized HTTP methods to enable the game client to interact with the server and access the scoring information. Two primary HTTP methods were employed for this purpose:

1. POST scores: The game client utilized the REST API's POST method to send player scores to the server. When a player completed a level or achieved a significant milestone, the game client would initiate a POST request to the server, transmitting the score data. This data was then stored securely in the server's database, ensuring the persistence of each player's score.

2. GET scores: The REST API's GET method was utilized by the game client to retrieve scores from the server. When required, the game client would initiate a GET request to the server, which would respond with the requested score information. This allowed the game client to fetch and display scores, rankings, or any other relevant gameplay data for the players.

By implementing the REST API with these HTTP methods, the game user could seamlessly communicate with the server database, securely transmitting and retrieving the scoring information. This integration enabled players to persistently store their scores, track their progress, and compare their rankings with other players.

The choice to deploy the Servers Database in Microsoft Azure provided several benefits. Azure's cloud infrastructure ensured high availability, scalability, and reliability, allowing for efficient storage and retrieval of gameplay data. Additionally, Microsoft Azure's robust security measures protected the integrity and confidentiality of the stored information, ensuring that player scores and ranking data remained secure.

4.7 Development lifecycle

The game was structured with distinct development lifecycle phases, which are outlined in **Table 1** for easy reference and understanding.

Phase	Process
Alpha	Alpha tests assure the first verification to confirm if the game's overall goals are met and that the game is effectively evolving into the predefined scope. These tests are performed by the project team developers.
Beta	Beta tests guarantee the first user experience of the gameplay, the feedback in terms of usability, playability, and the learning experience. These tests are by randomly selected users.
RC	Release candidate tests focus mainly on performance and heavy load tests. Once these tests are passed with success, assuring a real-life scenario load environment, the release candidate is cleared to be deployed or streamed to market.
RTM	The Release to the Market version of the game is the ultimate revised version in terms of bug fixing, with the highest degree of quality assurance. From this point on only bug fixing from end-user input is performed.

Table 1.
Game development lifecycle.

5. Game evaluation and quality scenario control

The evaluation of the project is conducted in accordance with recognized quality standards, specifically ISO 9126, which provides guidelines for assessing software quality, and SCORM, a set of standards for developing learning content. These standards ensure that the evaluation process follows established principles and best practices in the field.

To facilitate the evaluation process and ensure its objectivity and reliability, a quantitative evaluation framework called the Quantitative Evaluation Framework (QEF) is utilized. The QEF offers a systematic and structured approach to measuring the quality of the system in a quantitative manner. It operates at three levels: dimensions, factors, and requirements. Each dimension encompasses a collection of factors, while each factor consists of a set of specific requirements [8–10].

This work extends the foundational research conducted by Escudeiro et al. from 2008 to 2013 on QEF. By building upon their contributions, the researchers have further developed, adapted, and updated the framework to meet the evolving needs of evaluating educational serious games, demonstrating its ongoing significance in evaluating educational serious games [11, 12].

By employing QEF, the evaluation process becomes systematic and standardized. It enables the researchers to assess the quality of the system and to measurably evaluate outcomes, providing valuable insights into the system's performance.

5.1 QEF: dimensions, factors, and requirements

The quality scenario of the Elementals game is evaluated using QEF, which follows the standards set by ISO 9126 for quality assessment. The framework considers three key dimensions: Functionality, Adaptability, and Usability [8–10].

The Functionality dimension, which is vital for assessing the overall effectiveness of the game, consists of seven specific factors: gameplay, trading, collection, user interaction, features, account, and game engine. Each of these factors contributes to the overall functionality and performance of the game. Evaluating gameplay ensures that the game mechanics and interactions provide an engaging and enjoyable experience. Trading and collection assess the game's ability to facilitate the exchange and acquisition of cards. User interaction focuses on how players engage with the game interface and controls. Features encompass additional functionalities and enhancements that enrich the gameplay experience. Account evaluates the management and security of user accounts, while game engine examines the technical aspects and performance of the game's underlying system.

The Adaptability dimension assesses the game's ability to adapt to new conditions and requirements. This dimension consists of factors such as gameplay, versatility, accessibility, and maintenance. Evaluating gameplay adaptability involves determining whether the game can effectively adjust its mechanics and challenges to accommodate different player levels and preferences. Versatility assesses the game's flexibility in providing diverse gameplay options and modes. Accessibility examines how well the game caters to players with disabilities, ensuring they can fully participate and enjoy the experience. Maintenance evaluates the game's ability to receive updates, bug fixes, and improvements to ensure its continued functionality and relevance.

Usability, another crucial dimension, focuses on how well the game can be used by players to achieve their goals with effectiveness, efficiency, and satisfaction, as outlined by ISO 9241 standards. It incorporates factors such as menu navigation, map navigation, content quality, and integrity. Evaluating menu navigation involves assessing the intuitiveness and ease of navigating through the game's menus and settings. Map navigation examines the clarity and ease of navigating the game's virtual environment. Content quality ensures that the educational content, including cards and information, is accurate, informative, and engaging. Integrity focuses on the consistency and reliability of game data and ensures that players can trust the information provided by the game.

To evaluate each requirement within the dimensions, the QEF framework utilizes a discretized approach with predefined values. These values vary depending on the specific requirement. Some requirements are assessed using a binary approach, where they are either fulfilled (100%) or not fulfilled (0%). Others are evaluated using a Likert scale ranging from 1 to 5, allowing for a more nuanced assessment. Additionally, certain requirements follow a three-level threshold (0%, 50%, and 100%) or a five-level threshold (0%, 25%, 50%, 75%, and 100%) structure, which provides a graded evaluation based on specific criteria [8–10].

To ensure clarity and consistency during the evaluation process, all levels and thresholds for each requirement are clearly defined.

5.2 Evaluation methodology

The evaluation methodology employed by the project team involved assessing the product under development at various stages during the game's development process. The QEF provided a clear and quantitative perspective on the product's quality at any given stage.

This section outlines the approach to testing and data collection for the game Elementals, with a specific focus on its target population, including individuals with color blindness, and mobility and hearing impairments.

5.2.1 Technical testing

Technical testing is essential for ensuring product quality and goes beyond end-user testing. It provides insights into the system's capabilities, limitations, and performance issues, allowing necessary fixes to be made. The game specifications, including mandatory requirements and recommended specifications, are analyzed using the QEF framework to assess their completion and alignment with the project plan.

Performance testing involves evaluating the game on different devices with varying operating systems to assess compatibility and measure performance metrics. The goal is to identify any potential frame rate drops during gameplay.

5.2.2 Play testing

Play testing is a crucial process in game development where designers thoroughly test the game to identify and address bugs and design flaws before its release. Different types of playtest can be conducted. In this case, a playtest without external participants was chosen.

During the playtest, the researchers assess the game's usability by measuring how quickly users become familiar with the interface and navigate through the game. Feedback on game balance, mechanics (e.g., card picking, available options), and necessary improvements will be collected. The accuracy of the Dynamic Difficulty Adjustment (DDA) system will also be evaluated to ensure that the game's difficulty matches players' skills, regardless of disabilities. Analyzing the game's flow from early development stages helps streamline the process of fixing game mechanics.

5.2.3 Focus testing

Focus testing involves testing the product with a focus group to gain insights into people's feelings and opinions. In this type of testing, the group examines mockups and color schemes without actively using the product, understanding that their responses may change once they interact with the actual game.

The primary objective of focus testing is to evaluate the game's inclusivity for players with color blindness, hearing impairments, and mobility disabilities. The focus tests conducted include different testers:

- Color-blind: Testing the color schema with color-blind users to ensure that this impairment does not impact gameplay.
- Hearing-impaired: Testing the correctness and appropriateness of sign language used in the game for individuals with hearing impairments. It is essential to verify that sign language does not impact gameplay and is easily understandable.
- Movement impaired: Testing the user interface (UI) to assess its ease of use for players using the EPOC device or individuals with movement impairments. This test aims to verify if players with movement impairments can play the game without their disabilities significantly affecting gameplay.

5.2.4 Usability testing

Usability testing involves observing how individuals interact with the game and adapt to it if needed. Unlike focus testing, this method allows for the assignment of specific tasks and the collection of quantitative data.

During the initial interaction with the game, the researchers analyze players' responses to the interface, available options, gameplay, and the ease and efficiency of navigation. This testing phase aims to gather objective data on players' interactions with the game.

5.2.5 Data collection

The evaluation process relied on questionnaires and interviews as the primary methods for data collection. The choice of method depended on the limitations of the testers, with interviews being preferred for individuals with movement impairments and hearing impairments.

Questionnaires maintained consistency across different formats, whether written or verbal, and covered aspects such as user experience, graphics, gameplay, color schemes, and sign language adaptation.

5.3 Evaluation results

5.3.1 Alpha test analysis report

This section presents the findings of the analysis conducted on the Alpha Tests, providing an in-depth examination of the results to identify possible areas for improvement and assess the current state of the game.

Due to the previous Covid-19 restrictions, the researchers opted for an online questionnaire using the Microsoft Forms platform, facilitating both data collection and analysis. The Alpha Test included participation from 11 members of the development team, all of whom had no hearing impairments. However, it is worth noting that one question remained unanswered and unanalyzed as a result.

Overall, the evaluations for the questions were positive, highlighting aspects such as the ease of starting the game, text quality, game stability, interface design, avatar location, and ranking presentation. Despite the favorable feedback, there is still room for improvement, and certain considerations should be considered to achieve a more comprehensive assessment.

Specific improvements are related to the user interface, aiming to enhance navigation within the application. Conducting additional tests to identify potential errors and evaluate the game's performance is advisable. Furthermore, conducting a thorough review of the game's complete content, including text messages, color utilization, and other elements, is highly recommended. Lastly, assessing the effectiveness of the DDA as implemented in the game is crucial.

5.3.2 Beta tests analysis report

This section presents the findings of the analysis conducted on the Beta Tests, providing an in-depth examination of the results to identify possible areas for improvement and assess the current state of the game. Additionally, an evaluation based on the System Usability Scale (SUS) will be provided.

Similar to the alpha test, and to overcome Covid-19 restrictions, the researchers conducted an online questionnaire using the Microsoft Forms platform, facilitating both data collection and analysis. Once the questionnaire was distributed to the participants, they were instructed to play the game and respond to a predetermined set of questions. To ensure clear and consistent game testing, a comprehensive user guide was provided.

The Beta Test involved the participation of 11 subjects, all of whom had no hearing impairments. However, it should be noted that one question remained unanswered and unanalyzed due to this circumstance.

The evaluations for the questions were positive, with the notable aspects such as the ease of game initiation, text quality, game stability, interface design, avatar location, and ranking presentation receiving praise.

By employing the SUS analysis methodology, it can be concluded that the game has reached a usable and stable phase. Moreover, it is evident that all test subjects thoroughly enjoyed the game experience.

6. Conclusions

Education plays a crucial role in shaping the future of students, but it can often present unique challenges for those with disabilities. As such, the researchers aim to address these challenges by developing innovative educational approaches that ensure equity for all students. The researchers' primary focus revolves around the development of Educational Serious Games, which offer an inclusive and effective approach to delivering education. Specifically, their work contributes to this field through the creation of Elementals, an inclusive and educational game.

Elementals are designed to actively engage students and serve as a complementary solution to the traditional teaching and learning process. By harnessing the power of technology, the game promotes social inclusion and accessibility in educational settings, providing students with access to educational information and enhancing self-learning environments. The game particularly addresses the learning challenges in chemistry, with the potential to extend its scope to other subjects in engineering and electrical engineering education.

In future work, Elementals has the potential to expand its reach to encompass a broader range of subjects within the field of engineering and electrical engineering education. Through continuous development and improvement of the game, the researchers strive to enhance students' educational experiences and foster their learning across multiple disciplines.

Acknowledgements

This work is being developed at the research group GILT – Games, Interaction and Learning Technologies from the Polytechnic Institute of Porto, with the support of FEDER – Fundo Europeu de Desenvolvimento Regional through the Sistema de Incentivos à Investigação e Desenvolvimento Tecnológico – I&D Empresas, under the frame of Portugal 2020, project number POCI-01-0247-FEDER-069949, and the Erasmus+ Programme, European Education and Culture Executive Agency, through project InSign - Advancing inclusive education through International Sign, project number 2019-1-DE01-KA203-004964.

Conflict of interest

The authors declare no conflict of interest.

Author details

Paula Escudeiro* and Márcia Campos Gouveia
Games, Interaction and Learning Technologies, Porto Superior School of Engineering, Porto, Portugal

*Address all correspondence to: pmo@isep.ipp.pt

References

[1] Oliveira T, Escudeiro N, Escudeiro P, Rocha E, Barbosa FM. The VirtualSign channel for the communication between deaf and hearing users. IEEE Revista Iberoamericana de Technologias del Aprendizaje. 2019;**14**(4):188-195. DOI: 10.1109/RITA.2019.2952270

[2] Gagné RM. The Conditions of Learning. California: Holt, Rinehart, and Winston; 1965. p. 420

[3] Hackeloeer A, Klasing K, Krisp JM, Meng L. Georeferencing: A review of methods and applications. Annals of GIS. 2014;**20**(1):61-69

[4] Verma GK. Education and social integration for all: Challenges and responses. In: Approaches to Educational and Social Inclusion: International Perspectives on Theory, Policy, and Key Challenges. London: Routledge; 2017. pp. 9-21

[5] Michael D, Chen S. Serious Games: Games That Educate, Train, and Inform. Canada: Thomson Course Technology PTR; 2006. p. 313

[6] Carvalho CV, Coelho A. Game-based learning, gamification in education and serious games. Computers. 2022;**11**(3):1-4

[7] Gouveia D, Andrade A, Escudeiro P, Carvalho CV. Skill and competence development through games. In: Proceedings of the 16th International Conference on Information Integration and Web-based Applications & Services. New York: ACM (Association for Computing Machinery); 2014. pp. 49-53. DOI: 10.1145/2684200.2684350

[8] Escudeiro P, Bidarra J. Quantitative evaluation framework (QEF). Revista Ibérica de Sistemas e Tecnologias de Informação. 2008;**1**:16-27

[9] Escudeiro P, Escudeiro N. Evaluation of serious games in mobile platforms with QEF: QEF (Quantitative Evaluation Framework). In: Proceedings of the 2012 IEEE Seventh International Conference on Wireless, Mobile and Ubiquitous Technology in Education. Japan: IEEE; 2012. pp. 268-271. DOI: 10.1109/WMUTE.2012.65

[10] Escudeiro P, Escudeiro N. Evaluating educational games in mobile platforms. International Journal of Mobile Learning and Organisation. 2013;**7**(1):14-28. DOI: 10.1504/IJMLO.2013.051571

[11] Escudeiro P, Gouveia MC, Escudeiro N. Eco tetris: A serious game on sustainability. In: Proceedings of the 16th European Conference on Games-Based Learning. Lisbon: ACI; 2022. DOI: 10.34190/ecgbl.16.1.836

[12] Escudeiro P, Gouveia MC, Escudeiro N. ZooLector: Raising awareness of biodiversity loss to all. In: Proceedings of the 2023 IEEE Global Engineering Education Conference (EDUCON). Kuwait: IEEE. p. 2023. DOI: 10.1109/EDUCON54358.2023.10125247

Chapter 7

Education Supported by Games Even in University?

Kinga Kovácsné Pusztai

Abstract

Games are part of our lives and are not a waste of time because everyone needs intellectual experiences and development and the joy of playing. Multiplayer games play a role in the development of social connections too. Computer games made for educational purposes are called edutainment. Edutainment is part of gamification, which became known in 2010, and it means using game elements in other areas of life. In my chapter, besides providing some examples of mathematics and logic games and puzzles that proved to be successful when introduced in my university classes, I will deal with how to incorporate edutainment applications into university courses. In the broader spectrum of computer science, my research is focused on the subjects of "Algorithms and Data Structures" and "Programming methods" although some points and results can be applied to secondary school classes as well.

Keywords: gamification, edutainment, games and puzzles, computational thinking, IT education

1. Introduction

A game is an experience, a pleasure, a voluntary, free activity for pleasure where the driving force is the player's enthusiasm. There are numerous studies (e.g., [1, 2]) on the social benefits of the game. It plays a significant role in the development of social relationships. It builds character. It teaches the ability to win or lose, play by the rules, cooperate, and be self-disciplined. These are skills that you need in life.

The importance of the educational role of the game has been recognized since antiquity. The use of the game in the classroom has a motivating factor. It has a positive effect on the relationship between teacher and student, as well as on the student's active participation in the lesson.

Robert O. Brinkerhoff's study [3] has shown that participants in traditional training (methodology and environment) do not achieve sustainable behavioral change at 85% of participants. This means that 10 people out of 12 are unnecessarily involved in various pieces of training because the training is not followed by a lasting result. However, if the new knowledge of the learning process can relate to an appropriate experience, it strongly contributes to the deepening and later recalls of knowledge.

Today, games are increasingly being used successfully in the classroom (e.g., [4, 5]). Several concepts have emerged over the past decades to identify the use of digital games for educational purposes, but their precise definitions are not clear [6]. One frequently used term is "edutainment," which means short for entertaining teaching and learning. After the turn of the millennium, new terms such as "serious games" or "digital game-based learning" have spread. Both terms refer to the fact that the purpose of these games is to facilitate teaching-learning processes. Their aim is not to entertain, but that does not necessarily mean they are not enjoyable [7]. These applications are in many areas of life: industrial, military, scientific discoveries, healthcare, engineering, religion, politics, and, of course, formal education [8]. These digital games are interactive, based on a set of rules, aim to achieve some goal that challenges the player, and provide continuous feedback on progress in the game [9].

In my chapter, I approach games with two goals. First, I will deal with logic games, which can also be as programming tasks. In the second part of my chapter, I would like to show examples of how easy it is to create simple games, even for a theoretical subject, which is a great help to students in understanding, acquiring, and deepening their knowledge.

2. Games and puzzles in class work

Logic games are goal-bound and related to winning and competition. The game is a series of steps taken according to a well-defined set of rules, the goal and result of which is one or more precisely defined states, which means victory for one of the parties. In these games, chance plays no role at all. The outcome of the games depends more or less on the player's skill. In some games, there is only a winner or loser, while in others, draw is possible.

The goal of logic games is to help students to find a winning strategy. Students need to play with each other too, but one of the disadvantages of playing with each other is that students win not only when they find out the winning strategy but also when their partner plays badly. A well-programmed computer version of the game can help overcome this disadvantage.

Using logic games emphasizes experiential learning and observation skills that also develop problem-solving thinking, which we need more and more in an ever faster-changing world. According to Lénárd, a problem is any situation in which the way to achieve some goal is hidden from us [10]. Pólya, who was the creator of heuristics, defines problem-solving as [11] "The solution of any problem involves finding a way out of some difficult situation, circumventing some obstacle, reaching some goal that could not otherwise be achieved directly." The problem-based learning (PBL) is a teaching method and, according to some researchers (e.g., Walton, Matthews), a general teaching strategy that has been dealt with since the mid-twentieth century but has not yet been widely adopted in education [12]. The roots of PBL can be traced back to John Dewey, who believed that children should prepare for life. In traditional schooling, the child only learns knowledge but does not learn the most important thing: how to acquire knowledge. In life, however, there is no ready knowledge [13]. Problem-based learning is defined by Barrows as [14] "A conception of knowledge, understanding, and education that is profoundly different from the more usual concept underlying subject-based learning." Several articles (e.g., [15]) discuss the benefits of using PBL (e.g., motivating; or helping to develop competencies that are neglected in traditional education but are very important in life). An example of a problem might be determining the winning strategy of a logic game. Deductive reasoning plays a significant role in finding a winning strategy.

After some mathematical and logical games, students usually formulate conjectures and hypotheses that can be proved or disproved. The games also teach discipline and accuracy, which are needed in mathematics, programming, and real life [16].

Games and puzzles are typical forms of the interactive use of computers while displaying several qualities of computational thinking. Various problem-solving methods and heuristics of mathematics and information technology are inherent to any aptly compiled series of games and puzzles, hence their privileged role in modern pedagogy [17].

The aim of the logic games is not only to develop the student's intellectual abilities but also to spend their free time in a meaningful and enjoyable way, to socialize, to strengthen sociability, to compete regularly, and to foster traditions [18].

Logic games are also an excellent tool for teaching programming. I have used the games presented below in the Programming Methods course of the BSc in Computer Science Teacher training. This course was in the first semester and aimed at acquiring programming skills. As an input, no prior knowledge of programming is required. Therefore, the examples presented are applicable not only in higher education but also in public education.

2.1 Some logic games

2.1.1 Tower of Hanoi

Perhaps one of the best-known and most widely used games in education is the Towers of Hanoi, which is used to teach recursion. The exercise, which is also known as the Towers of Brahma, or the doomsday puzzle, is as follows [19]: *put all the discs on the first rod onto the second one. There are two important rules: (1) you can only move one disc at a time, and (2) no bigger disc can get on top of a smaller one.*

This game was invented by Édouard Lucas, a French mathematician, in 1883. His inspiration was a legend that at the time of the creation of the world, Brahma's monks started playing a 64-disc version of the *Tower of Hanoi* game. According to the legend, when the monks are finished stacking the discs onto the second rod, the monastery collapses, and our world ceases to exist.

For the precise formal specification of the task, we can introduce the *Hanoi (n, i, j)* predicate in which *n* is the number of discs, *i* is the source, and *j* is the destination of the move. The original problem then becomes formalized as *Hanoi* (64, 1, 2).

The solution has the following steps (**Figure 1**):

If there is one disc, lift the disc from rod I and put it onto rod j, or in other words, use the function *move* (*i*, *j*).

If there is more than one:

Move the top *n-1* discs from the first rod to the third one: *Hanoi (n-1, i, n-i-j)*.

Move the largest disc from the first rod to the second one: *move (i, j)*.

Move the top *n*-1 discs from the third rod to the second one: *Hanoi (n-1, n-i-j, j)*.

Playing through the game, increasing the number of discs one at a time, students will understand the idea of induction. If we explain how a previous iteration can serve as a model for calculating with an increased number of discs, the idea of recursion is also introduced and understood. The solution illustrates the "divide and conquer" principle in program design along with recursion. Summarizing, by using this game, skills like algorithmic thinking, inductive inference, generalization, and abstraction are developed.

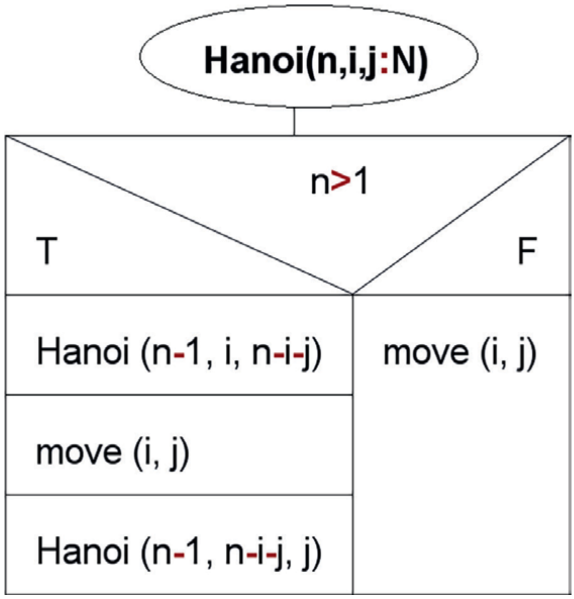


Figure 1.
The structogram of the solution of the game.

2.1.2 Guess my number

Almost all of us know the game that starts with the following question: *Guess the number I have thought between 1 and 100. With at least how many questions can you guess for sure?* The idea behind the solution is halving intervals, which is an excellent way to demonstrate the theorem of logarithmic search. The solution will be 7 because $\log_2 100 = 6{,}6 < 7$.

Perhaps a more difficult and less familiar version of the task is to *guess the number by writing down all the questions in advance and having no control over the order in which the guesser looks at the questions and answers them. Are seven questions enough in this case, or do we need more questions?*

This task can be answered if we produce some cards in advance, and our questions are whether it is on the given card. If we make the cards skillfully, then 7 questions are still enough. But how do we create our card so that we can guess the number by looking at it? The binary number system is used for the solution. The first digit of each card will be a power of two. The card contains the numbers whose binary number is one in the given position. In this way, the thought number is obtained by adding the first numbers of the cards for which we received the answer that the thought number is on the card. (The cards are shown in **Figure 2a**.)

Making the cards can be a simple programming task. In this case, the guessing game is not limited to 100, but to n. We do not use the binary numeral system when creating the program. It is easier if we notice some regularity about the numbers on the number cards. (The algorithm for creating the cards is shown in **Figure 2b**.)

The task can be made even more difficult by leaving the world of numbers and choosing another topic. I attempted to do this in the Challenge online competition that was aimed at primary school students and also my first-year university students.

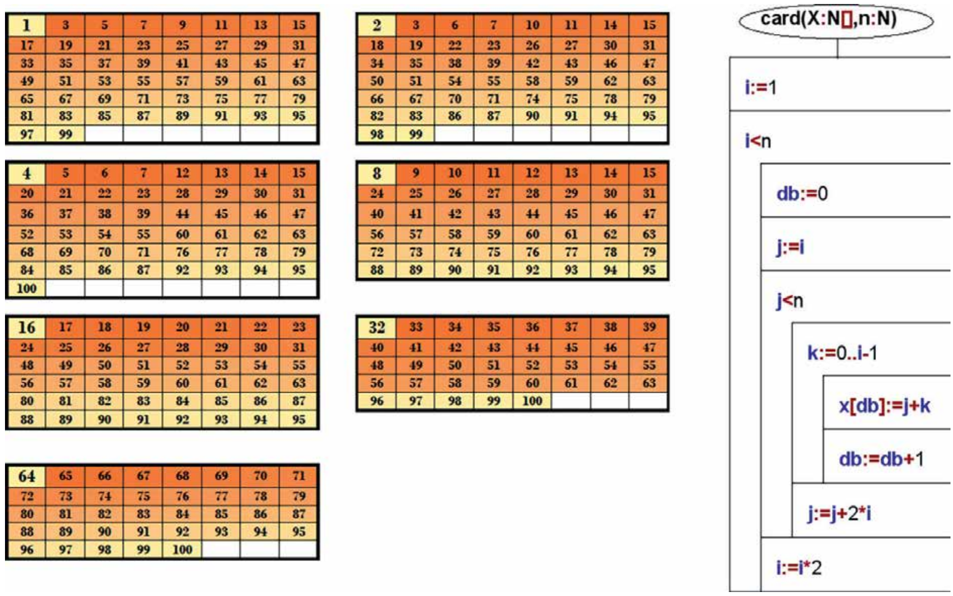


Figure 2.
The cards and the structogram of creating the cards.

The chosen topic was Olympic sports. The best answer to 5 questions listed 25 sports. (The solution is shown at the following link: https://people.inf.elte.hu/kinga/bar-kochba.png.).

3. Gamification and edutainment

Computer games made for educational purposes are called edutainment. Edutainment, a portmanteau of the words "*education*" and "*entertainment*," refers to technologies and software products that combine education with entertainment in some way. In the digital age, many of these products and technologies seek to make education more attractive to young people and students [20].

Educational technology takes many forms. A streaming video platform, a prepackaged learning product, or an app for a mobile phone can be classified as edutainment. Such instructional applications are very effective in practicing tedious routine tasks (e.g., in mathematics operations with fractions) or in very complex concepts (such as the operation of an ecosystem). Therefore, edutainment can be a significant issue in the development of modern digital and hybrid learning materials for classroom and supplementary educational use [20].

Edutainment is a subset of gamification. The concept of gamification in the public mind appeared in the 2000s and has become popular since 2010, but its origins can be traced far back. Fuchs [21] found examples of the use of gamification in the army dating back to early Roman times.

The most accepted definition of **gamification** is *"the use of game elements in a non-gaming context"* [22].

Two very important concepts appear in the definitions of gamification: game elements and game mechanisms, which are often called game design techniques. Game elements refer to tools taken from traditional and video games and game mechanisms to the application of the operating principle.

Of course, tools will work effectively only if the mechanisms of the game are given: A game is a voluntary, promising success, transparent, and properly delimited (providing proper time).

The "out-of-game context" term in the definition implies that the purpose of the game is different from that of gamification. The biggest difference between game and gamification is that a game is an activity providing entertainment or amusement, whereas the purpose of gamification is to do something to achieve a predefined goal in real life.

3.1 The appearance of edutainment in "Algorithms and Data Structures" practical university course

In a previous semester, I supplemented my practical lessons with edutainment applications. These included Kahoot quizzes that I usually used for summarizing. As it was so popular with students, last semester I decided to create a Kahoot quiz for each of my classes. I continued this initiative this semester as well, and in addition, I also introduced the use of Kahoot quizzes to check class attendance. I find the Kahoot repetition at the beginning of the lesson very useful as it is easy to use and does not take too much time, but the students like it. It spices up the lesson, briefly summarizes what they learned in the previous lesson, and confronts the students with their gaps, and after using them, the students tend to pay more attention. In addition, it sends me feedback on the success of the answers to each question and the student's performance. To make the game more interesting, the students who solve the most questions correctly get 1 extra point, which is added to their test score.

Kahoot is a well-known and popular quiz-maker application that I presented in detail in my previous work [23]. I will therefore not describe its use here. However, based on my experience, I recommend taking the following aspects into account [23].

- First, I consider it important that a Kahoot quiz is not too long and does not want to repeat too much knowledge. (A quiz for me consists of 5–7 questions.) A Kahoot that is too long becomes boring, takes a lot of time out of the lesson, and contains more information than the students can remember, so it is not effective.

- It is important to pay attention to the time limit. If the question is too long, then more time should be allocated, but by overestimating the question and allocating more time than necessary, the competitive spirit is reduced and thus the students' motivation.

- One of Kahoot's drawbacks is that it shows the images in a small size. An image is at its largest when it is lying flat, and its aspect ratio is closest to 350 × 220.

I researched the students' feedback using anonymous questionnaires that were filled out by 62 students. (17 of them were English-speaking and 45 were Hungarian-speaking students.) Its result can be seen in **Figure 3**.

The question "How much do you like that we repeat with Kahoot at the beginning of the class?" was rated on a five-point scale with an average of 4.44, a standard deviation of 1.07, a mode, and a median of 5. (The Hungarian students rated it with an average of 4.6, while English students rated it with an average of 4).

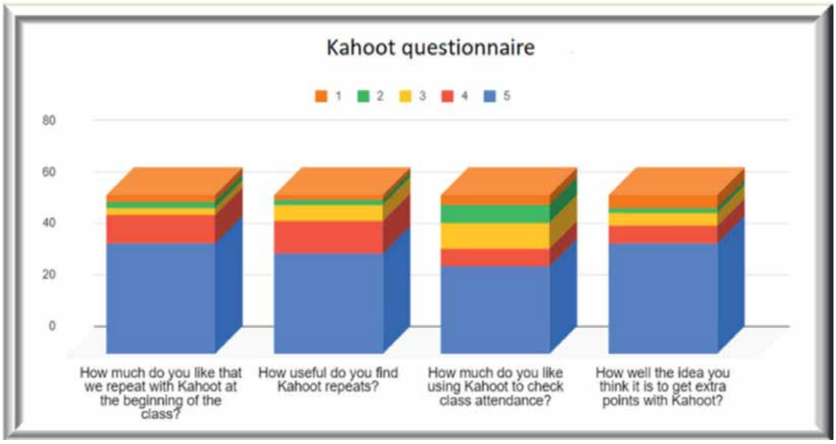


Figure 3.
Students feedback on the Kahoot.

The usefulness of the Kahoot repetitions was similarly rated, with an average of 4.37 (Hungarian students scoring 4.53 and English students 3.94), a standard deviation of 1.01, a mode, and a median of 5.

The idea that checking class attendance with Kahoot was rated with an average of 3.97, a standard deviation of 1.33, a mode, and a median of 5. This was the only question that the English-speaking students rated well (4.11) than the Hungarian-speaking (3.91) students.

And finally, the idea that students could earn extra points with Kahoot didn't appeal to the students as I thought it would. They rated it with an average of 4.31, a standard deviation of 1.25, a mode, and a median of 5.

The students had the opportunity to write down their opinions. I received a total of 20 text opinions (that's about a third of the students), which I was very happy about as it meant that almost a third of the students thought it was important to evaluate them textually as well. Some opinions about it: "*It's a really smart choice to take attendance with Kahoot because in this way we don't lose time for it, furthermore we can get extra points*;" "*I think it's a great way to repeat what you've learned in the previous class, rather than just starting the next material;*" "*Kahoot gives extra motivation to prepare for class after class;*" "*Fun and cool* ☺."

Based on student feedback, I will use Kahoot in my other classes to check class attendance.

3.2 The appearance of edutainment in "Algorithms and Data Structures" lecture university course

My previous research was about the gamification of practical lessons. Based on my own and others' experiences, the use of gamification elements has a positive effect on student motivation, which leads to a deeper knowledge of the material.

My current research is about the gamification of lectures. In contrast to practical classes, lectures are more difficult to activate students, as we have 300 students at a time instead of 20. If we use a presentation, the vast majority of students do not take notes, as the presentation is usually downloadable. For this reason, several of my colleagues returned to traditional blackboard lectures, but this did not bring the appropriate student activity either. Unfortunately, many students complain that the explanation is told to the board, so they do not hear much of it, and it is also difficult to write down the material, so it is even more difficult to pay attention to the explanation.

A further problem is that the lecture material is generally more theoretical than the practice, with more definitions, theorems, and proofs, which makes it much harder to hold the student's sustained attention. Many students fall behind in the first half of the lecture and from then on do not pay attention or try to re-engage with the lesson. Since the assessment of the subject is an exam during the exam period, students do not even take much care to understand it during the study period. And the backlog is increasing more and more, so it takes much more time to master the knowledge during exam time. This is why I started to expand the course with edutainment applications.

I have created an edutainment application for each lesson which is optional. These apps are short games purposed to help you review and understand the lesson material. As they are not long, they are not suitable for learning the whole lesson, but they are excellent for highlighting, reinforcing, or illustrating them with examples of some of the most important knowledge of the lesson. A further advantage of its brevity is that the highlighted parts are just so long for the students to remember. I hope that the short games are motivating enough for them to engage with it so that they come to the next lesson with some prior knowledge that will help them to understand more of the lecture.

My longer-term goal is that when learning for the exam, they will already have some prior knowledge to help them learn. Because they will not be unfamiliar with the material, they will be able to master it more easily and more deeply in less time.

Of course, there will be students who will play these games during the exam period. In this case, the mid-semester benefits will not be applied, but they will still help with the learning process.

When creating the games, I tried to use as many images as possible, thereby favoring students with a visual style. I think that today's world is becoming more and more visual, so there are more and more people who can understand the material more easily with the help of a picture. (These images are usually also included in the lecture.)

The app for the first lecture was viewed by 55 people in the first week, which means 17.2% participation in the 307 people's lecture. Of course, this rate is not entirely true, as some of the 307 students did not attend the lecture (due to class collision or exemption), and there are certainly students who opened the app more than once. The game for the second lecture was viewed by 44 people; it means a 14% attendance rate. However, the viewing of games does not stop after a week; currently (at 2/3 of the semester), these apps are at 123 (40%) and 128 (41.7%) views. All these numbers may not seem so high, but considering that there is no exam period yet, there is no assessment yet, so there is no need to learn the material yet, I think these are not bad rates.

I was curious to know the students' opinions that I collected through an online questionnaire. A total of 119 students filled in the questionnaire and rated the games on a five-point Likert scale with an average score of 4.52 (with a standard deviation of 0.7, a mode, and a median of 5) (**Figure 4a**). So far, 114 out of 119 students (95.8%) have been helped by the apps in their learning (**Figure 4b**). The students could also write opinions about the games, some of which were: *"It playfully teaches useful basics. It is very good, we should have similar games in more subjects;" "It helped me to review not*

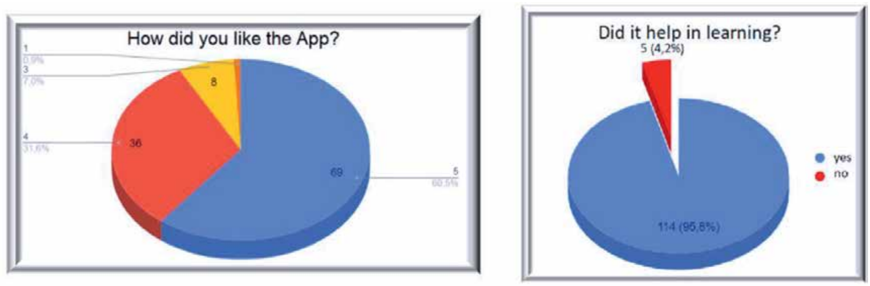


Figure 4.
Students feedback on the apps.

only the practice but also the lecture notes;" "I liked that it was specific to the course material." Text responses also revealed that some students do not attend the lecture and do not read the notes, but open the game and then watch the presentation of the lecture to solve the game.

As I introduced my innovation this semester, I do not know yet how it will affect my exam grades.

In this semester, I will make 12 games for the lecture. In the following, I will present some of these that I have already created.

3.3 Presentation of some edutainment applications

Nowadays, there are a lot of free interactive curriculum maker applications that are very easy to use. (Some of which I have presented in Edutainment in Education article [24]).

I created my edutainment applications using LearningApps. I like it because LearningApps.org is a versatile tool for teachers. LearningApps.org is a Web 2.0 interactive multimedia application, to support learning and teaching processes with small interactive modules. We can use it without registration, but after free registration, we have much more control over our blocks (called Apps) and our classes [25]. It is very easy to use with a walkthrough tutorial on the home page explaining all of the features. I can create your activities from scratch using one of the templates provided or you can adapt activities already created. There are games and quizzes such as matching, identifying, categorizing, gap-filling, crosswords, filling in answers, ordering, putting things on a line, and multiple-choice tasks [26].

3.3.1 The millionaire game

In the first lecture, we covered several topics. In addition to introducing basic concepts and notation, we dealt with the time complexity in more detail and started the topic of sorting. "The Millionaire Game" app is a great opportunity for repeating this type of lesson.

The App is a 6-question quiz, embedded in a game. The difficulty is that with a wrong answer, you lose and must start the game from the beginning. The home screen (**Figure 5b**) shows 6 values, each of them hiding a task. There are four possible answers to a question (**Figure 5a**), of which only one is correct. If you find the correct answer, return to the home screen, where the diamond moves up one value; that is, you continue to the next question. If you give a wrong answer, the correct answer can

Figure 5.
A question of the millionaire game, and the home screen.

Figure 6.
The end of the millionaire game by losing, or by winning.

be seen, but the game is lost (**Figure 6a**); that is, it ends. The game is won if all the questions are answered correctly (**Figure 6b**).

I was afraid to create "The Millionaire Game" type app because I thought that it would frustrate the students if they made a mistake on the last question. But it does not frustrate them; it motivates them to solve the task correctly. Meanwhile, going through the first questions several times, they unconsciously learn the answers.

According to the questionnaire, this game was one of the most popular, with students rating it 4.62 on average (they only rated it with fives and fours), and it helped all students but one to learn. The game received some text comments, some of which were "*The gamified approach is fun;*" "*My little boy enjoyed it and ask me to search for tasks on the LearningApps that were suitable for him;*" "*I think it's a good idea because, in case of a mistake, the repeated questions help to memorize it.*"

As the feedback showed that the students liked this type of game, I also created "The Millionaire Game" app for a future lecture.

3.3.2 Group assignment

To practice elementary data structures, I created a group assignment game. The app exercises the knowledge of the array, the queue, the simple one-way linked list, and the one-way list with a header node. At the beginning of the game, these four groups appear in different colors (**Figure 7a**). The cards (24) are visible in the middle, which you need to drag into the right group. If the cards run out (or if you want to stop gaming), the solution can be checked by clicking on the checkmark in the blue

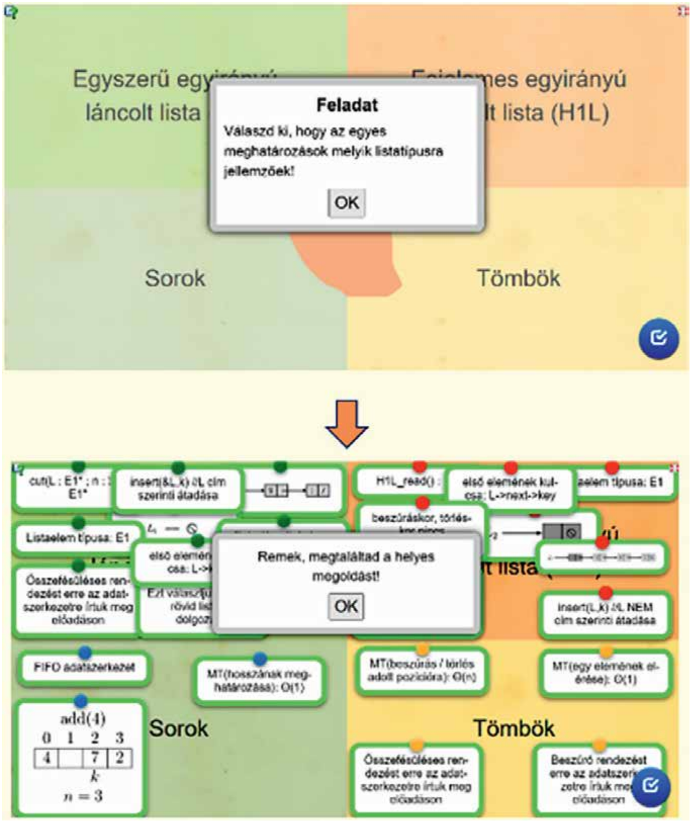


Figure 7.
Group assignment start and end screen.

circle in the lower right corner of the screen. The correct solutions are given a green frame; the incorrect ones are given a red frame (**Figure 7b**).

Some cards contain images; others contain important properties of the data structures (e.g., time complexity, when to use, and advantages over other data structures).

This is the game that most students dealt with so far. In the questionnaire, students rated the game with an average score of 4.51, and all said it helped them to learn. The app received some text comments, some of which read like this: "*I liked that it was specific to the curriculum. I would highlight that the (previous) answers were visible throughout;*" "*It was useful, I think it was a good summary of the similarities and differences between the data structures.*".

Feedback from students confirmed that this type of app is useful because it is easy to give a visual example, which is increasingly important for today's generations. Based on the feedback, I will decrease the number of the cards.

3.3.3 Matching pairs

To practice the two-way lists, I created a matching pairs game. One-half of the pairs is a picture, and the other half is an explanation of the picture. There are eight pairs in total; two pairs represent the types of two-way lists, and six represent the structograms of its operations. As the images are quite small, clicking on them will enlarge them so

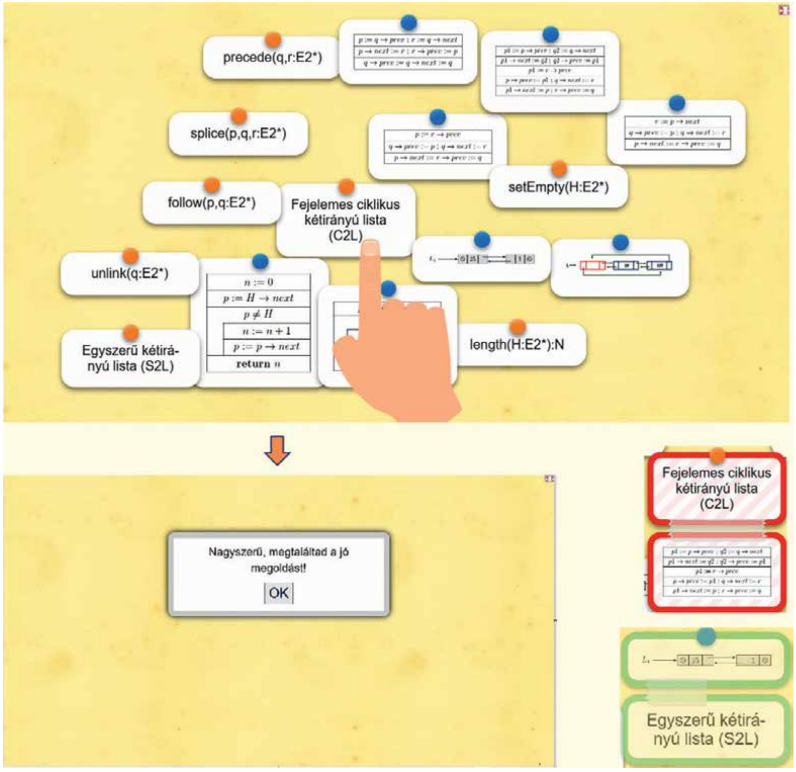


Figure 8.
Matching pairs start and end screen and an example of a bad and a good pair.

that they are visible. Initially, all the cards are on a virtual corkboard, with one-half of the pairs with an orange angle and the other half with a blue pin (**Figure 8a**). The cards can be matched in any order, starting with the easiest one you know for sure and finishing with the hardest one. When marching, you must take one of the cards to its pair. If you did well and found a pair, then the members of the pair are linked, given a green frame (**Figure 8d**), and then disappear from the board. If not, the cards get a red frame (**Figure 8c**); then, you can take them apart with one click to continue the game. The game is over when all the cards have been matched; that is, they have disappeared from the board. A congratulatory message (**Figure 8b**) will then be received.

This game has been rated the most and in the most different ways. The average rating was 4.29 with a standard deviation of 0.96. The app also received some text comments, one of which reads: *"I liked that the concepts and their explanations were visualized in one place, at the same time." The game also received a constructive opinion: "I liked it, I think it would be even better if we could see all the parties together after the successful solution."*

Based on the feedback, I modified the app afterward, so you do not get immediate feedback about the pairing, but your pairs do not disappear in the end (**Figures 9** and **10**).

3.3.4 Crossword

To practice the basic concepts of the binary tree, I created a crossword puzzle with 13 questions, 4 horizontally and 9 vertically. The question can be solved in any order,

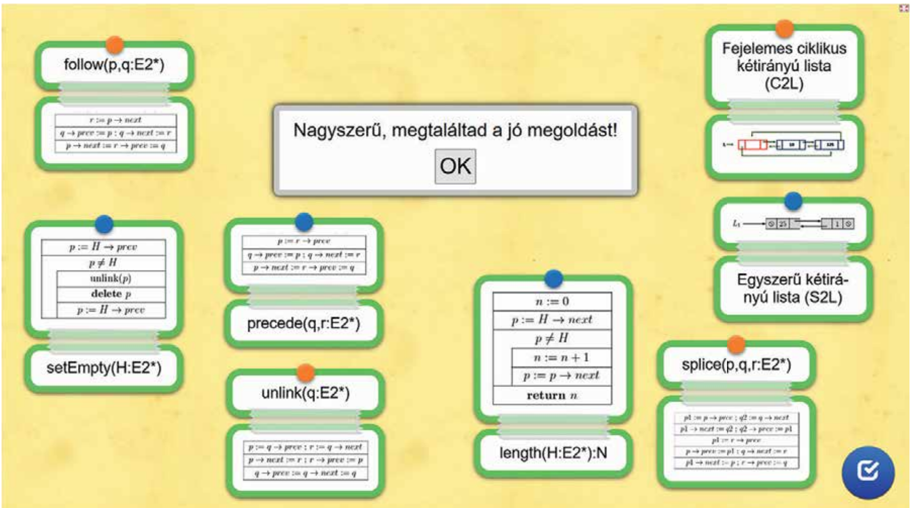


Figure 9.
Matching pairs: Game over for good solutions.

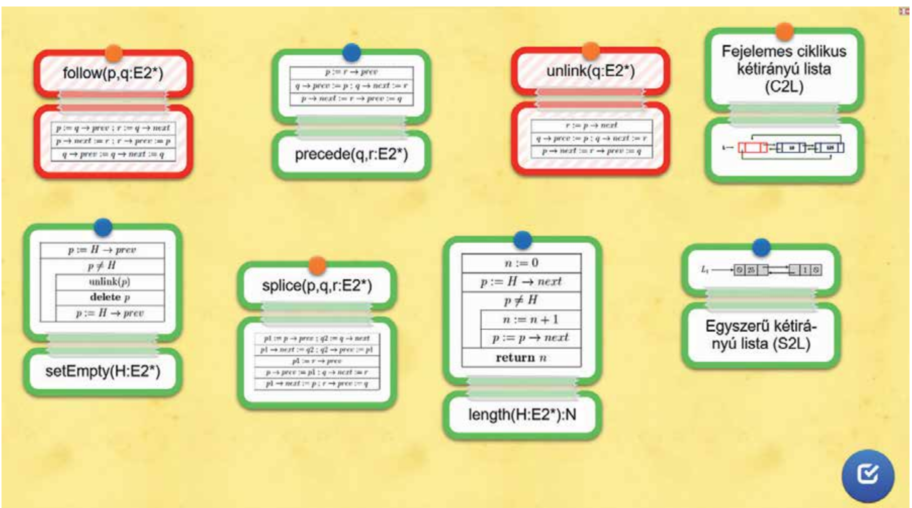


Figure 10.
Matching pairs: Game over for bad solutions.

with the help of the letters already filled in (**Figure 11b**). When checking, the letters that are filled in correctly turn green, and the incorrect ones turn red (**Figure 11c**). If you have solved the crossword correctly (**Figure 11a**), a new word as a solution will be received.

Based on the questionnaire, this game was the most successful, with an average score of 4.8. The student only rated it with fives and fours, and it helped all students to learn. It is clear from the comments that the students liked the app. ("*I enjoyed this one, I liked the tree in the background and the questions made sense, but you had to know the concepts.*")

Feedback from students confirmed that this type of app is useful, because playful enough for students to take out their notes and review important concepts. I plan to

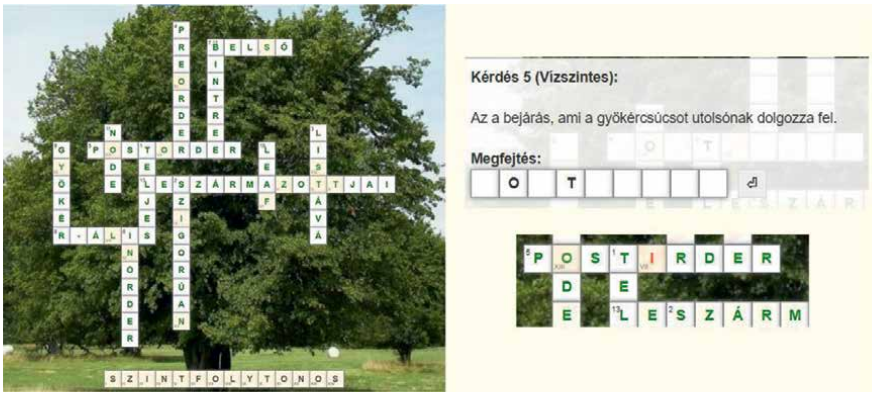


Figure 11.
Crossword well-filled end screen, a question, and an example of a bad solution.

innovate similarly in my other subjects in the future, and based on the feedback I have received, I will use this type of app there as well.

4. Summary

Experience-based learning builds on our curiosity as a natural component of our human operation. The drive is a very strong urge to accompany us from our childhood. There may be a lot of hindrances in the learning process, such as lack of motivation or disinterest. However, using games in teaching can turn our knowledge into a problem-solving action; then, we can be a success [27].

In my chapter, I approach games with two purposes.

First, I have dealt with logic games that are suitable for programming tasks. The featured games are examples of how logical games can use to teach programming. There are a vast number of games that can be used in secondary school or university education. Although I have approached the application of games with a programming focus, the scope of the application can be extended; that is, they can be used in other areas as well.

The presented games are not new, but they can still be new to many students as they have not necessarily seen them before in this digital age. These games can be applied with great success in teaching certain concepts by helping the development of problem-oriented thinking.

In the second part of my chapter, I wanted to show examples of simple games that can be created, even for a theoretical subject, which can greatly help students to understand, master, and deepen their knowledge. This section is mainly of pedagogical importance; although I have shown examples within the areas of computer science, the methods I have presented are not only related to computer science but can be applied to the teaching of any subject.

In this section, I have created some edutainment applications for the "Algorithms and Data Structures" university lecture and practice courses, and I also have researched the student's opinions.

The high participation of students in the survey and their feedback confirmed that there was a place for the use of gamification not only in public education but also in higher education.

Author details

Kinga Kovácsné Pusztai
ELTE IK, Budapest, Hungary

*Address all correspondence to: kinga@inf.elte.hu

References

[1] Frost J, Worthamés S, Reifel S. Play and Child Development. 4th ed. New Jersey: Pearson; 2011

[2] Vygotsky LS. Play and its role in the mental development of the child. International Research in Early Childhood Education. 1967;**7**:6-18

[3] Brinkerhoff DRO. Telling the story of program impact with compelling evidence [Online]. 2018. Available from: http://www.brinkerhoffevaluationinstitute.com/ [Accessed: October 20, 2021]

[4] Larsen MCK, Aline O, Frey Peter M, Dolan Robert P, Victoria V, Aaron MV. A Literature Review of Gaming in Education. London: Pearson; 2012

[5] Young MF, Slota S, Cutter AB, Jalette G, Mullin G, Lai B, et al. Our princess is in another castle: A review of trends in serious gaming for education. Review of Educational Research. 2012;**82**(1):61-89

[6] Susi T, Johannesson M, Backlund P. Serious Games-an Overview. Sweden: Technical Report HS-IKI-TR-07-001, School of Humanities and Informatics, University of Skövde; 2015

[7] Pásztor A. Digitális játékok az oktatásban. Iskolakultúra. 2013;**23**(9):12

[8] Kankaarantaés MNP. Design and Use of the Serious Game. New York: Springer; 2009

[9] Tobias SAFJD. What research has to say about designing computer games for learning. Educational Technology. 2007;**47**:20-29

[10] Lénárd F. A problémamegoldó gondolkodás. Budapest: Akadémiai Kiadó; 1984. p. 37

[11] Pólya G. A gondolkodás iskolája. Budapest: Gondolat; 1977

[12] Newman M. A pilot systematic review and meta-analysis on the effectiveness of problem based learning. Medicine, Dentistry and Veterinary Medicine. Special Report. 2003;**2**:77

[13] Dewey J. Democracy and Education: An Introduction to the Philosophy of Education. New York: Macmillan; 1916

[14] Albaneseés M, Mitchell S. Problem-based learning: A review of literature on its outcomes and implementation issues. Academic Medicine. 1993;**68**(1):52-81

[15] Szögedi I. A probléma alapú tanulás, mint új gyakorlati készségfejlesztő módszer, az egészségügyi felsőoktatásban. Pécs: Pécsi Tudományegyetem Egészségtudományi Doktori Iskola szerk; 2012

[16] Oláhné Téglás I. Logikai játékok szerepe a matematikaoktatásban. Módszertani Közlemények. 2015;**55**:4

[17] Moursund D. Introduction to Using Games in Education: A Guide for Teachers and Parents. Oregon: Teacher Education, College of Education, University of Oregon; 2006. p. 97403

[18] Nagy EK. A logikai és táblajáték-foglalkozások szerepe a matematikatanításban. Új Pedagógiai Szemle. 2007;**57**(6):88-92

[19] Wikipédia, Hanoi tornyai, Wikipédia [Online]. Available from: https://hu.wikipedia.org/wiki/Hanoi_tornyai [Accessed: April 13, 2023]

[20] Rouse M. Edutainment. Panama: Technopedia; 2016

[21] Németh T. English Knight: Gamifying the EFL Classroom. Piliscsaba: Pázmány Péter Katolikus Egyetem Bölcsészet- és Társadalomtudományi Kar; 2015

[22] Deterding S, Sicart M, Nacke L, O'Haraés K, Dixon D. Gamification. Using game-design elements in non-gaming contexts. In: Proceedings of the 2011 Annual Conference Extended Abstracts on Human Factors in Computing Systems. Canada: ACM; 2011. pp. 2425-2428

[23] Pusztai K. Innovative Methods in Computer Science Education. London: IntechOpen; 2023

[24] Kovácsné Pusztai K. Edutainment in education. In: New Methods and Technologies in Education and Practice. Trnava: XXXII.Th DIDMATTECH; 2019. p. 13

[25] Horváth J. Learningapps.org vagy Kockalapok.hu - IKT módszertani ajánlat. Budapest: POK; 2016

[26] Swabey MT. The Digital Teacher [Online]. 2017. Available from: https://thedigitalteacher.com/reviews/learningapps [Accessed: May 12, 2023]

[27] Zichermannés G, Linder J. The Gamification Revolution: How Leaders Leverage Game Mechanics to Crush the Competition. New York: McGraw-Hill; 2013

Chapter 8

METs as Gamified Health Indicator to Promote Elderly Active Lifestyle and Technology Acceptance in Ambient Assisted Living

Xavier Fonseca

Abstract

This paper focuses on ambient assisted living (AAL) scenarios and proposes the use of location-based games (LBGs) as engaging applications for (1) the promotion of an active lifestyle in healthy senior adults (+65) and (2) the enhancement of current acceptance rates of technology used in these scenarios. It offers a high-level software architecture that can be used to integrate health indicators produced from gameplay data of LBGs with AAL healthcare systems, thus serving as data sources capable of contributing to better professional healthcare support. The proposed concept enables care providers in AAL settings to recommend gaming exercises that can be done through LBGs; in turn, such professionals have access to health indicators (metabolic expenditure) of the gameplay, which can then be compared to the WHO recommendations for an active lifestyle of older adults. This concept enables the use of digital LBGs running on commonly available smartphones without the need for extra hardware, as applications that are more engaging and motivational than traditional technologies by design. A test of concept for the proposed architecture is presented, whereby the health indicator METs are offered from multiple gameplay data provided by an LBG and where such indicator is compared to dedicated hardware.

Keywords: location-based games, ambient assisted living, health, concept, architecture

1. Introduction

Ambient assisted living (AAL) explores technology to support the ageing process of the elderly and chronically ill. This is a concept that contributes to a more sustainable national healthcare system, through which high costs and stress in the healthcare infrastructure can be alleviated [1]. Within AAL, technology is designed and used to turn ageing into an easier and (to some extent) self-dependent process to individuals, all the while aiming to prevent, heal, and improve overall wellness. Even though being a promising concept, researchers have focused on the technical evolution of AAL systems [2–4] (e.g., sensors [5, 6], IoT-based devices and architectural designs [7], dependability analysis [8], complex systems engineering [9], modeling of

AAL-related factors [4, 10], and artificial intelligence in AAL [11]). Less effort has been put on user acceptance of AAL technology [2, 12]. This, together with the status quo of legacy systems of formalized care institutions that tend to be substantially sized, complex, and closed [13–15], prolongs the lack of adoption of AAL systems in the larger-scale national healthcare systems [3].

Pervasive serious games are applications that are designed early on to engage players and that can serve as an alternative approach to user acceptance. The serious games discussed in this paper are digital location based, that is, games that run on location-aware smartphones connected to the internet and that are designed to provide an outdoor pervasive gaming experience. Even though argued to be lacking overall social acceptance themselves as "serious" applications for healthcare [2], serious games in general are established as valid options for multiple health-related purposes such as rehabilitation [16], physical exercise [17–19], and cognitive training [20]. Serious location-based games (LBGs) serve as a potential means for a more sustainable and engaging health improvement approach to AAL settings, as they expose healthy players to their physical surroundings and motivate them to go outside and do fun game activities to advance gameplay. When the serious game is an active LBG, movement (and maybe even physical exercise) must be made to progress through the gameplay. The activities promoted by LBGs can be purposefully designed to be beneficial for the elderly living in AAL scenarios and must be researched further to have their potential properly unlocked in AAL settings.

This paper presents a concept of LBGs for healthy senior individuals living in AAL locations, a concept that informs researchers and practitioners on how to integrate LBGs in health and wellness applications used by healthcare professionals in these locations. The presented concept proposes the benefit of combining LBGs with AAL technology, by proposing LBGs as data sources for health indicators that are usable in AAL scenarios. It offers an instantiation of this new architectural concept with the existing LBG "Secrets of the South," which offers one health indicator that is relevant for AAL scenarios: the estimated metabolic expenditure (METs) of the gameplay sessions of that LBG.

The following section of this paper provides a brief background on the evolution of AAL systems and their current maturity level and the use of engagement and LBGs as key research avenues to improve current technological approaches in AAL scenarios. The next section proposes the health indicator MET as one appropriate indicator for older adults, all the while proposing a conceptual architecture, where LBGs work together with AAL systems to inform them on the metabolic expenditure during gameplay sessions throughout the week. The section after that instantiates this concept with the LBG Secrets of the South. The following section discusses the presented proof of concept and highlights implications for game development and computer science. The last section concludes with future work.

2. Background

2.1 Ambient assisted living and its maturity level

Several factors, such as innovation in technology, the demand of citizens for easier and updated access to healthcare, and a justified concern over rising costs and stress in existent formalized care infrastructure, have driven the pursuit of new healthcare solutions and models [21, 22]. These have sought to transform the present reactive healthcare system centered on treating patients' ailments, into a user-centered care provision centered on what users (instead of patients) want. Among multiple transitions, the concept of AAL emerged as a model based on the connected devices over the internet (IoT), which aims at promoting an active and assisted self-centered ageing process to the elderly and chronically ill primarily in their home [23]. AAL is the use of IT technologies (AALTs) for the enhancement of the quality of life of the elderly via active and healthy applications. The AAL concept (1) refers to an ecosystem that (2) is enabled by cloud and IoT technologies in particular [24] and that (3) promotes health and wellness applications for a high-quality ageing process. This ecosystem includes the elderly's relatives, social services, health workers, and care agencies [25].

Current state of the art on AAL is most solidly based on the technological aspects of AAL scenarios, leaving other aspects such as user acceptance of such new technology largely unaddressed [25]. Over 95% of the research published within the last 3 decades on AAL [26] falls within 12 clinical themes [27]: routine action monitoring (42.7%), fall detection (13.7%), physiological parameters tracking (8.6%), presence detection (7.7%), gait analysis (6%), assessment of environment (6%), sleep monitoring (5.8%), estimation of level of activity (4.3%), routine support (1.7%), gesture recognition (1.5%), indoor localization (1.5%), and wandering study (0.4%). This body of research has different focal points, mainly on the health status (e.g., asymptomatic seniors, dementia, or Parkinson's), use case settings (home, smart homes, nursing homes, hospitals, public places, and other settings), population (elderly, younger target groups, caregivers, and healthcare professionals), different technologies (e.g., sensors for contact, monitoring, motion, presence, pressure, or temperature), and methodological perspectives (it is predominantly quantitative [27]).

The purpose of AAL is to improve and promote an active quality of life enjoyed by the elderly, but it is largely lacking the measurement of the impact of these new technologies in the user's life. Current aspects such as user acceptance, based on perceived usability, ease of use, intention to use, and actual use [28], show that seniors currently perceive AALTs as useful, but they are hesitant to accept and adopt them [28–30]. This means that there is a mismatch between proposed AALTs and what works for the elderly and care personnel alike [30] and shows that AAL as a concept is still materialized as a set of prototypes that do not offer extensive and much needed qualitative insights for AAL to become a consolidated reality.

2.2 User acceptance of AAL technology: engagement and the contribution of location-based games

Given the presented foci on technological aspects, one potentially beneficial but largely missing research direction in AAL literature is the design for engagement. Engagement will hardly solve structural and technical problems hindering the daily use of AALT such as data security issues or the extra workload imposed on care professionals [31–33]. Yet engagement provides an ever-evolving set of design strategies that can lead to engaging user experiences and higher adoption rates of technology [34, 35]. These strategies vary at least per target group, as each has distinct capabilities, needs, and preferences [36]. The elderly in particular have a set of conditions that younger generations do not: beyond the expected lack of IT proficiency, they experience a decline in cognitive, physical and sensorial functions, and may face some impairment that is unique from the already diverse age group they belong to [37].

Numerous strategies exist to influence the level of engagement of the elderly with interactive technology in AAL settings, such as the allowance of creative expression, emotional attachment, knowledge sharing, and activities with a purpose they identify with [17, 38, 39]. It is worth noting, however, that by "the elderly", this paper refers to healthy independent senior individuals and not to those seriously impaired mentally or physically. This is an important positioning when arguing about engagement because the observed lack of engagement of seniors with technology in AAL settings may be due to issues far deeper than the potentially benevolent lack of motivation to use technology [28, 40].

This paper defends that engagement is a missing design goal for technological adoption in AAL scenarios and proposes serious games as one type of application that must be engaging by design to benefit AAL systems. Serious games should be designed to be engaging [39–42], and the study of the impact of serious games in the elderly's engagement can contribute to the adoption of AALTs, for example, on subjective qualitative experiences commonly not accounted for in the AAL literature, physiological reactions, motives for playing, game usage, time spent on playing, and the impact of playing on life satisfaction [43]. Existent research on serious games for health is abundant [19, 44–48], yet research on the combination of serious games for AAL scenarios is rarely done [2, 49]. The only works on serious games and AAL are the ones developed by the human-computer interaction center at Aachen since 2015 [2, 49–51]. Brauner, Wittland, and Ziefle have researched the use of serious games for cognitive training [2, 51], physical exercise [2, 49, 50, 52], and social acceptance [2] and focus on providing design guidelines for the successful introduction of serious games for health purposes to residents of AAL environments. They discuss the social acceptance of serious games and argue that this is an emergent field with challenges at (1) the relationship level between game performance and intention of use, (2) user diversity, and (3) the lack of IT literacy of seniors. Thus, with this exception, serious games for AAL settings are hardly researched [2, 49].

This paper argues that LBGs, as a specific type of digital serious games that run on devices with locative features and data connection, contribute to the state of the art of applications for AAL settings and influence user acceptance of AALTs. LBGs as complement applications to AAL environments are an unexplored research avenue, and if they are properly designed for a prolonged and sustainable user engagement, they (1) promote an active gameplay experience that is commonly outside AAL settings and (2) offer a user experience that can serve as potential motivator for a more active lifestyle to healthy seniors. LBGs are a game genre that promotes outdoor play around public space, and this is largely not considered in AAL research possibly because researchers have the assumption that the elderly must be assisted and hardly have autonomy. LBGs offer clear benefits when compared to traditional AAL applications (and even regular serious games): when the elderly are healthy and have the autonomy to walk outside their living location in an unrestricted way, they can actively move around while engaged in play. When the game design matches the knowledge of what the elderly prefer playing, and have the physical ability to play, LBGs may serve as an extra motivator to a more active lifestyle when compared to the scenarios where they are permanently located within the AAL facility.

3. Concept proposal for healthy elderly (65+) in AAL locations

For serious games to be relevant in AAL scenarios, they must contribute to the overall AAL technical system in place, and this paper proposes a conceptual model, where LBGs provide health indicators that complement AAL systems. This paper proposes the use of metabolic equivalents (METs) as an estimate of the amount of physical activity done by the elder. With this health indicator (derived from multiple data sources from the smartphone where the LBG runs), a conceptual architecture is presented where LBGs are integrated with an AAL technological environment and where they work together to promote an active lifestyle with technology.

3.1 Metabolic equivalents as a health indicator of active lifestyle for the elderly

The metric MET is a physiologic estimation of absolute exercise intensity and bases itself on the amount of energy that an individual's body consumes per minute while resting [53–56]. It varies based on the intensity of the activity performed: light activities (<3 METs/min) include slow walking or light gardening; moderate activities (3–6 METs/min) include faster walking or leisure bicycle; and vigorous activities (>6 METs/min) include running or faster active sports [57]. METs are an important estimate measure because they are a reliable estimate of effort that can be used to understand how the individual is doing regarding the recommendations of the world health organization (WHO). WHO recommends several guidelines to children, adolescents, adults, and older adults on physical activity, which, in general, recommend a minimum amount of time for moderate to vigorous-intensity physical activity and a reduction of sedentary lifestyle.

Specific to older adults (+65), the WHO argues that the benefits of physical activity are critical in many situations such as adverse events, all-cause and specific cause mortality, cognitive outcomes, falls and fall-related injuries, functional ability, frailty, osteoporosis, and mental health. As part of their weekly physical activity, seniors should do varied physical activity of moderate to higher intensity (>= 3 METs/min) for at least 3 times a week to enhance functional capacity, and it is recommended to all older adults and to not just those with reduced mobility [58]. In terms of METs, the minimum recommended by WHO is 450 METs per week.

Given that the MET is an estimate measure of physical effort, it is appropriate for scenarios where dedicated sensing hardware is not deployed (e.g., electrodes attached to the person's body), like those where only a smartphone is used. In LBGs, it is important to provide a playful experience that is fluid and does not offer a big barrier to start playing (like the setup of body sensing hardware) [18, 59], which makes the estimation of METs a compatible approach.

3.2 Location-based games as media to Gamify METs and promote usage of technology in AAL settings

LBGs can serve multiple purposes in AAL settings, such as serving as engaging digital applications to seniors and serving as applications that source relevant data to both AAL settings and formalized healthcare systems. In terms of the sort of data this paper refers to, LBGs can calculate the amount of METs that players consumed every time they played the LBG and give back the aggregated weekly amount of METs to the AAL system. This can inform healthcare professionals on the seniors' level of physical activity outside the AAL location.

To that end, this paper proposes the conceptual architecture of **Figure 1**. This conceptual architecture assumes a 3-tier AAL system as what is commonly proposed as future healthcare systems: with edge and fog computing as architectural concepts needed to adopt new IoT sensors/actuators and handle the consequentially huge amounts of data with a short reaction time [24, 60]. The architectural proposal for the LBGs is based on what these games require to work in terms of infrastructure and third-party services (e.g., map provision, gaming accounts, and geolocated points of interest) [17, 61].

Such architectural concept for LBGs further specifies a separate storage for health indicators processed by the game, and this is a security measure by design to only allow access to this database by the AAL system (instead of the whole game data). The conceptual systems architecture in **Figure 1** presents a connection between the AAL location, other healthcare institutions responsible for the electronic health record data of the elderly (e.g., multiple hospitals or clinics), and the LBGs. AAL systems should have access to the personal EHR of care recipients because each person is unique from the clinical perspective, and this information, recorded over a lifetime, should be considered by AAL entities. It is admissible for AAL systems to pass data onto formalized healthcare systems not related to LBGs, so the bidirectionality is represented.

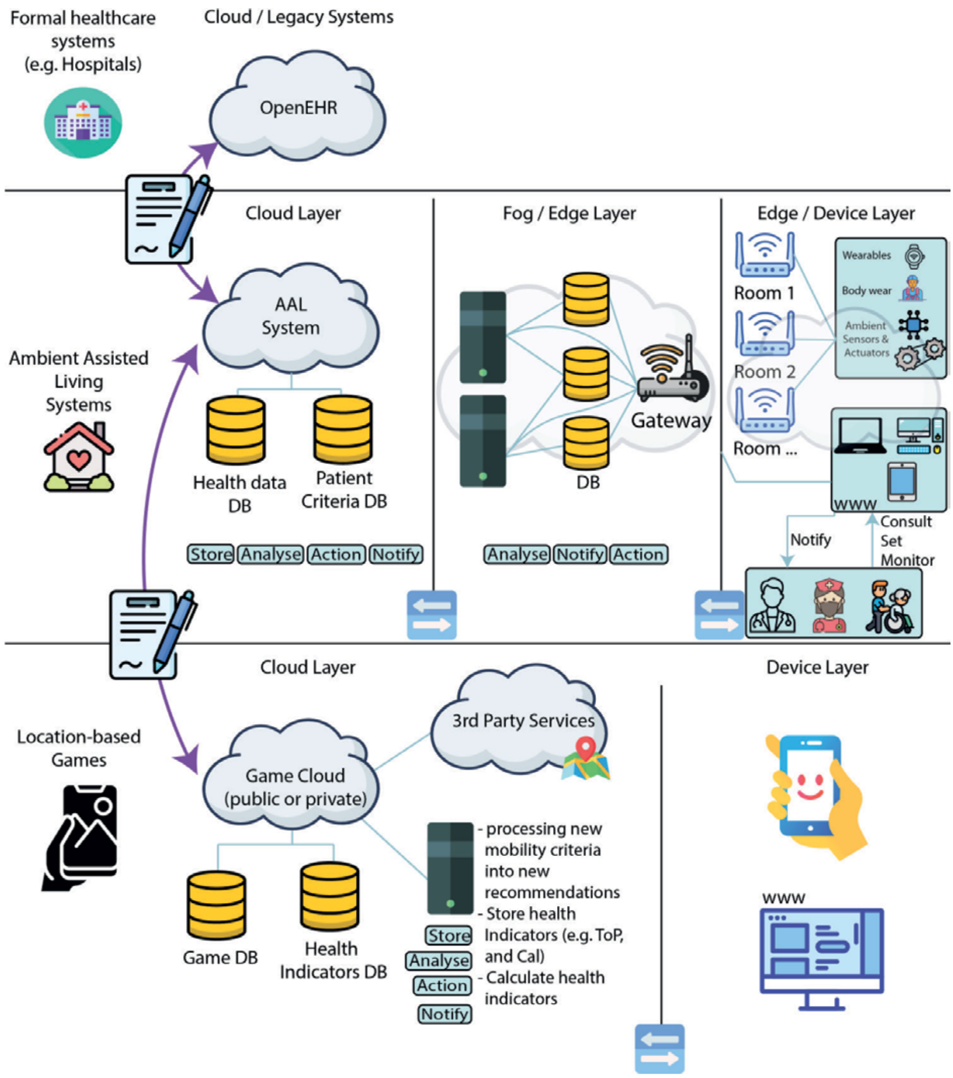


Figure 1.
Conceptual architecture to use LBGs as data sources for AAL settings.

Parallel to formalized healthcare systems, AAL systems are also integrated at the cloud level with LBGs, with two main data flows. The first one is the establishment of patient recommended activity criteria by the AAL care staff, which can inform LBGs of the desired goals from the AAL perspective. The second one pertains to the access of relevant gameplay indicators from the AAL care personnel that contribute to the overall health data collection that AAL systems collect.

Regarding the 1st identified dataflow, the proposed concept establishes the use of criteria for activities that each older adult can/should perform. These criteria are defined by the care staff of the AAL environments and are, in turn, consumed by LBGs to propose differentiated gameplay experiences to seniors. Every time the LBG is played, the game consults such criteria defined and stored in AALTs and recommends gaming activities based on the updated criteria. The admissible criteria are not specified in this paper but may include: (1) the type of physical activity allowed or level of risk the senior has; (2) a scale recommending more movement (in METs); (3) more cognitive training (e.g., memory exercises); and (4) more activities promoting emotional gain (e.g., giving back to society, having random social interactions, or transmitting knowledge).

Regarding the 2nd identified dataflow, LBGs store and process data from the game state and gameplay. This goes from GPS coordinates to game-related events that players encountered during gameplay. Much of this data can be post-processed in the game cloud into valuable health indicators that are in turn available to be included in the AAL care providers' assessment on how active and healthy each OA has been. Specifically on health indicators, LBGs can calculate the amount of METs spent through the following game indicators:

- Activity type (**Activity**): The computing of energy expenditure is directly associated with the type of activity performed (e.g., walking, fast pace, running, or jumping).

- MET level of Activity (**MET Level**): The computing of energy expenditure is directly associated with **Activity** and the level of effort applied. This is the MET per full minute of activity of a given activity type, and the different METs/min per type of activity (and the vigorousness involved) are found here.[1]

- Distance (**GPS**): what distance was covered while doing the activity.

- Time of Play (**ToP**): within which amount of time the given activity was done.

The two mentioned data flows of the presented concept happen across different systems on the cloud (those of the AAL and the LBG). These systems work completely autonomously by definition: LBGs can work without an integration with AAL systems, and vice versa. The flow of data must therefore be explicitly agreed upon between each elder of the AAL location and their LBG account: only they may decide to enable their gameplay as a data source for the analysis and monitoring of their health and wellness in the AAL settings.

With these game indicators, the amount of METs can be computed first by finding the speed with which the type of activity was done:

[1] http://media.hypersites.com/clients/1235/filemanager/MHC/METs.pdf, Levels of Common Recreational Activities, Wellsource, Inc, 2008.

$$\text{Speed}(\text{Km}/\text{h}) = \text{GPS}/\text{ToP} \quad (1)$$

With **Activity** and the speed from Eq. (1), the **MET Level** can be found here[1]. With that information, the amount of METs of each LBG gameplay can be found with the following equation:

$$\text{METs} = \text{MET Level x ToP} \quad (2)$$

All these indicators (**Activity**, **MET Level**, **GPS**, **ToP**, and **METs**) are to be stored in "Health Indicators DB" of **Figure 1** and can then be used to inform healthcare professionals on the weekly amount of METs spent by each patient of an AAL location.

4. Test on MET conceptual validity

From the conceptual architecture presented in **Figure 1**, this paper presents a test of concept where an existent LBG is used to provide an estimate of the amount of METs from a gameplay session and where such estimate is compared to the effort measured through a dedicated smartwatch. This is to reflect on the soundness of the health indicators estimated through LBGs played on a smartphone, not to validate the concept.

4.1 Context: the LBG secrets of the South

Even though not designed specifically for users aged 65+, the SotS is chosen to further explore the proposed concept. It is an LBG that the authors have information and control over its implementation; it fits the proposed concept and can be used to generate and test the estimation of METs as a health indicator.

The LBG Secrets of the South (SotS) is a game created for the promotion of meaningful social interaction in public space and has been validated with both adolescents and adults in both Rotterdam and the Hague, the Netherlands [17, 18, 39, 42]. It is a game that proposes gaming activities (named challenges) spread across the city, and these activities require different sort of actions to be solved: Quizzes with open and closed answers, taking pictures and voting on other players' pictures, doing physical activities together in a group of players (e.g., playing music or football in the middle of the street), or performing given actions within a time limit [18, 36, 41, 62]. Players can play this game by opening it on their smartphones and looking on the map to see the nearest ones. Given that the game is originally designed to promote social interaction with passers-by, the game activities proposed usually invite players to engage with other people on the streets, such as interviewing them, inviting them for a dare challenge, or asking for their collaboration toward a given task (e.g., high fiving a given number of people within 2 minutes). On top of the described gameplay, players can also go to the online game portal and propose game challenges for other players to do in their own neighborhood. The game architecture of the SotS LBG is shown in **Figure 2**.

Figure 2 provides an example on how to implement **Figure 1**'s LBG system: the SotS gaming application requires many services such as the recording of the game state, the recording of the position of players, account authorization, and the usage of 3rd party services such as the geographical data with points of interest of the

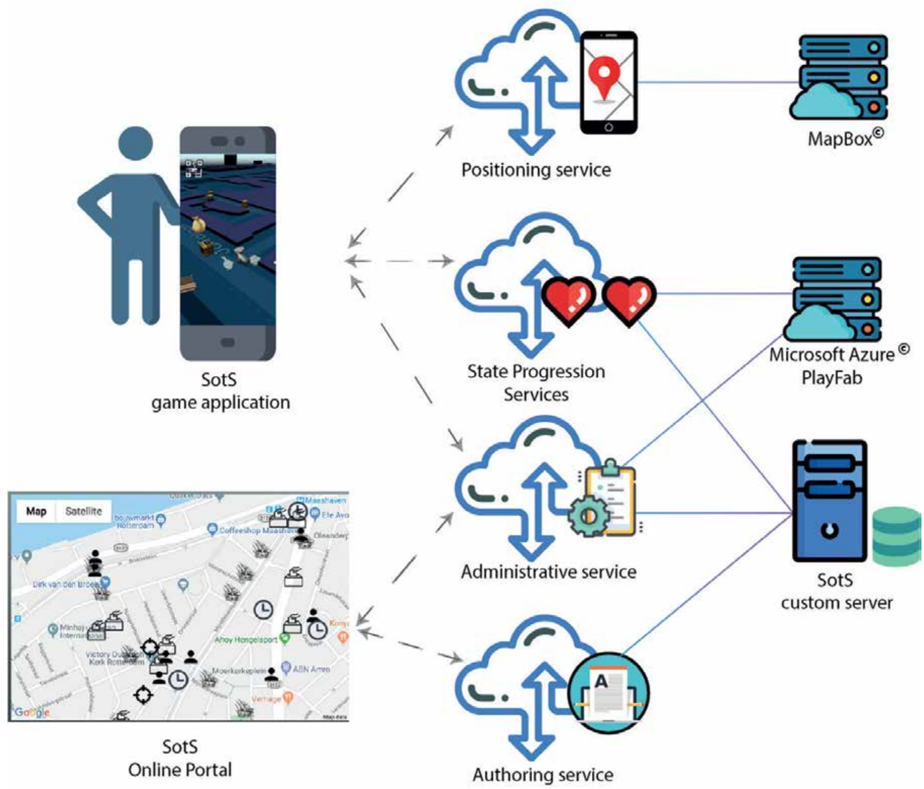


Figure 2.
The SotS software architecture [17].

surrounding area of the player. On the cloud of the game, a custom private server with storage capabilities ("SotS Custom Server") is implemented, where both the "Game DB" and "Health Indicators DB" from the presented conceptual systems architecture in **Figure 1** are built in.

4.2 Gameplay test session and health indicators with the SotS

A gameplay test session was prepared with the SotS to exemplify how, from an LBG gameplay, the indicators GPS (distance) and ToP (time of play) are made available to then estimate the amount of METs from a gameplay session. This test session served to test the idea of using LBGs in AAL settings as proposed in the concept above and not to validate the presented concept. It started with the design and set up of several gaming activities around a random public place accessible by an older adult, which were defined in the SotS online platform as shown below:

The gameplay test session was executed with one person to cover most of the challenges placed around a given public space in Porto, Portugal, and see how health indicators could be derived from the SotS game data. **Figure 3** shows several 2D icons on the map representing the different types of gaming activities created in the SotS game. The test session lasted around 25 minutes in a walking pace, with a senior male individual weighing 78kg, and the resulting gameplay data that were logged by the LBG are plotted in both **Figures 4** and **5**.

Figure 3.
Setup of game challenges on public space.

Figure 4.
Data collected from the gameplay test session with the SotS. Yellow pins represent collected GPS coordinates by the SotS; green arrows make the locations of the gaming activities previously defined in **Figure 3**.

Figure 5 shows a small snipped of the entire game data logged by the SotS, which are needed for the functioning of the SotS (other LBGs may define other data). From these data, the indicator time of play (ToP) and GPS (distance) are directly calculable by reading the game database (column "timestamp" in **Figure 5** for ToP and columns "latitude" and "longitude" for distance). In this game test session, ToP was 25 minutes, and total distance walked was 1.68 kms. This distance was walked (this is the type of activity).

From the collected data, we can use formula (1) to calculate the speed of walk, which results on an average 4 km/h. That means that, by checking the MET level of

latitude	longitude	timestamp	message
41.17638016	-8.605589867	2021.07.30 14:22:25	
41.17642975	-8.60534668	2021.07.30 14:22:31	
41.17645645	-8.605324745	2021.07.30 14:22:36	Otis opened the Quiz Challenge "Who's the new cool kid?" [XsfhCZeFfRS7G37ds]
41.17646408	-8.605313301	2021.07.30 14:22:37	
41.17638016	-8.605481148	2021.07.30 14:22:43	
41.17638016	-8.605481148	2021.07.30 14:22:50	
41.17629623	-8.605381012	2021.07.30 14:22:54	Otis SOLVED the Quiz Challenge "Who's the new cool kid?" [XsfhCZeFfRS7G37ds] with
41.17629242	-8.604920387	2021.07.30 14:23:29	Otis closed the Quiz Challenge "Who's the new cool kid?" [XsfhCZeFfRS7G37ds]
41.17629242	-8.604920387	2021.07.30 14:23:29	Otis opened the Multiplayer Challenge "UPTECH nice fellas" [C6jYDoDNsb7K5CHi4]
41.17629242	-8.604920387	2021.07.30 14:23:29	Otis opened the Multiplayer Challenge "UPTECH nice fellas" [C6jYDoDNsb7K5CHi4]
41.17629242	-8.604896545	2021.07.30 14:23:31	
41.17629242	-8.604838371	2021.07.30 14:23:34	Otis closed the Multiplayer Challenge "UPTECH nice fellas" [C6jYDoDNsb7K5CHi4]
41.1762886	-8.604784966	2021.07.30 14:23:38	
41.17630386	-8.60472393	2021.07.30 14:23:44	
41.17632294	-8.604580879	2021.07.30 14:23:50	
41.17632294	-8.604477882	2021.07.30 14:23:57	
41.17633438	-8.604352951	2021.07.30 14:24:03	

Figure 5.
Snippet of the entire LBG game data collected by the SotS: latitude and longitude (GPS coordinates), timestamp (data/time of record), and message (game event logged).

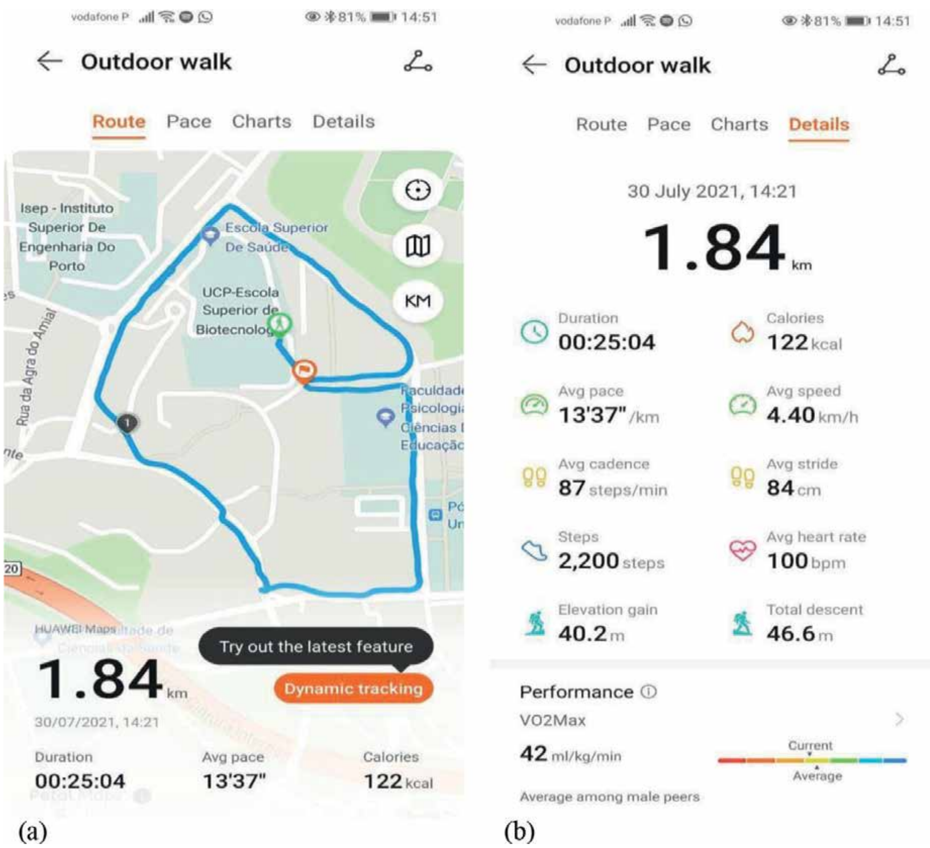


Figure 6.
Data collected from the wearable during the game test session: (a) path recorded as 1.84 km, (b) detailed view of the physical activity.

walking at an average walking speed of 4 km/h[1], it is possible to find that the MET level is 2.9 METs/min. Using the formula (2), this results in 72.5 METs total spent walking during the gameplay session (around 99 kcal, based on the formula provided and the weight of the participant).

A wearable was used[2] to compare real data from this gameplay test session (**Figure 6**) with the values estimated by the game SotS (from **Figure 5**). Based on the smartwatch, the distance traveled is 1.84 km instead of 1.68 km (observable by the blue line- **Figure 6**a), with the details of the walking session in **Figure 6**b. Of these, it is relevant for the comparison with the LBG gameplay the distance walked, average speed, duration measured, and the kilocalories measured by the equipment.

Comparably to the data gathered by the game itself, the distance captured is more detailed. This means that we can find a speed of 4.4 km/h with formula (1), which translates into MET Level of 2.97 and results on a total amount of 74.25 METs. This closely compares to the 72.5 METs/min estimated by the LBG SotS. In terms of calories, the 99 kcal estimated through the data of the gameplay compare to 122 kcal estimated through the wearable.

5. Discussion

This paper presented the concept of location-based games (LBGs) for the promotion of an active lifestyle in AAL scenarios, a concept that informs researchers and practitioners of not only how to benefit from LBGs in AAL settings but also on how to integrate LBGs as data sources in existent AAL systems. This integration refers to the use of LBGs as producers of health indicators, specifically the use of gameplay data such as the type of **Activity**, **MET Level**, distance (**GPS**), and time of play (**ToP**) to calculate the quantity of metabolic equivalents (**METs**) spent during such gameplay. These METs estimated by LBGs are an estimate measure of absolute exercise intensity that can help care providers invite older adults to be more active and allow them to follow up on exactly how well seniors respond to an increased active lifestyle. Parallel to the extraction of health indicators from LBGs, the proposed concept aims at contributing to a higher acceptance rate of technology in AAL settings in healthy seniors, when compared to traditional applications. This may be the case when the employed LBGs are engaging and well designed for this target group.

The test on METs validity was made with one senior individual and with an existent LBG not originally designed for this scenario or target group. Given that, and the number of players used in the test, the previous section cannot and is not meant to validate the concept. It serves to illustrate that the metabolic estimate calculated by the LBG SotS is close (≈ 19 kcal difference) to what can be derived from dedicated hardware used to produce the equivalent estimate. This indicates that the use of LBGs to promote physical activity can also serve to provide an informative and reliable estimate of the metabolic expenditure of the gameplay. In this case, the proposed conceptual framework would be able to inform the AAL healthcare professionals that this senior participant spent exactly 72.5 METs over 25 minutes in a potentially engaging application, which could then be compared to the World Health Organization's set of recommendations (or others) at the end of the week to further support the wellbeing of the older adult.

[2] The wearable used was a Huawei smartwatch GT2, paired to the Huawei Health application running in parallel to the SotS game.

5.1 Implications for computer science and game development

This case study delves into the complex information systems that are deployed both at formal (hospitals) and informal healthcare sites (e.g., home or elder houses) and argues toward a specific health indicator through gameplay mediated by LBGs. The first implication of this is that health-relevant indicators such as metabolic equivalents can be extracted from gameplay-related indicators. This not only poses as a guideline for game designers of such games but also takes a step forward toward reducing the documented lack of social acceptance of serious games as valid applications for healthcare. Going even beyond the social acceptance of serious games, health indicators that are produced by systems such as LBGs may be seen by hospital staff as being worth less than that which is produced by certified hospital equipment. This may breed the need to create mechanisms in IT healthcare systems of formal care institutions to legitimate and accommodate data from third-party sources such as LBGs. It may also imply an extra rigor from the software development's perspective to guarantee that health indicators produced by LBGs in AAL settings are a source of information that healthcare physicians can rely on.

A second implication is the use of LBG as a driver toward the evolution of the current healthcare IT systems. The future of healthcare is argued to go from a reactive "I've got a disease" care system to a proactive/preventive "I want to take care of my own health" care system [22]. This means that healthcare information systems will be different in the future: (1) they will be centered around what the user (instead of patient) wants and (2) they will enable more personalized healthcare by potentially accounting for information produced by third-party non-medical software or somehow release part of the patient's information to other systems controlled by the user. This serves to guide the evolution of current healthcare IT systems toward higher interoperability and flexibility. At the same time, the use of METs by LBGs as posited by this chapter contributes to enabling this future of healthcare while at the same time considering user acceptance of technology.

Lastly, another implication for game development and healthcare systems is that there is the possibility of having to certify location-based games as medical devices in the future so that their data can in fact be accounted for in wider healthcare IT systems. This means designing and developing location-based games where health-related data produced is secure, private, and locked behind mechanisms; only the user can articulate. Such a mechanism can be the release of information (METs produced by an LBG) only when the doctor asks the patient to or when the user explicitly gives order for the game to upload the information to the AAL information system. Even though this may pose as a usability challenge for the elder, it may enable the integration of LBGs as data sources in existing healthcare IT systems and support the user-centered care paradigm.

6. Future work

Regarding future research, the conceptual architecture should be further explored with use cases that can implement LBGs and AAL systems in AAL locations. First, engagement levels that LBGs can achieve in older adults must be researched and how to design for a sustainable and prolonged gameplay that most likely occurs around the same places over time. This is a known research issue in games in general, which are particularly relevant for LBGs. Second, future research needs to further the state of the art on technology adoption: do engaging LBGs influence user acceptance of AAL technologies (AALTs)? Third, future research should assess whether the health indicator considered in this paper (METs) is the only or the most useful metric for the discussed scenario, which breeds the need to perform lengthier studies on how well/reliable are the METs produced form LBGs. Fourth, future integrations of LBGs and AALTs must enforce privacy (and pseudo-anonymity) by design. It is conceivable to imagine scenarios where seniors do not want their health care providers to access their gameplay information but still want to play the LBG. Other topics are more concerned to games themselves, such as the sort of content that is appropriate to this target group and who manages and maintains such application in the longer run. Lastly, the literature on AAL ecosystems must be further developed from the user-centered design perspective, as the elderly are usually not expected to be fully literate IT users. Seniors should have a dignified ageing process, and technology must seamlessly aid in the process and serve as neither an impediment to health and wellness nor as a tool excluding individuals from social interactions mediated by technology.

Insights on these issues can have positive implications that far outreach AAL systems and have implications at the larger-scale national healthcare systems of current societies.

Conflict of interest

No conflicts of interest are declared.

Author details

Xavier Fonseca
CIR, ESS, Polytechnic of Porto, Porto, Portugal

*Address all correspondence to: fxsf@ess.ipp.pt

References

[1] Cicirelli G et al. Ambient assisted living: A review of technologies, methodologies and future perspectives for healthy aging of population. Sensors. 2021;**21**(10):3549

[2] Brauner P, Ziefle MJJ. Social acceptance of serious games for physical and cognitive training in older adults residing in ambient assisted living environments. Journal of Public Health. 2021;**2021**:1-13

[3] Spoladore D et al. A semantic-enabled smart home for AAL and continuity of care. In: IoT in Healthcare and Ambient Assisted Living. Springer; 2021. pp. 343-371. DOI: 10.1007/978-981-15-9897-5_17

[4] Gião J et al. Profiling based on music and physiological state. In: I-ESA'16: Interoperability for Enterprise Systems and Applications. I-ESA 2016. 2016. DOI: 10.1007/978-3-319-30957-6_10

[5] Maskeliūnas R, Damaševičius R, Segal S. A review of internet of things technologies for ambient assisted living environments. Future Internet. 2019;**11**(12):259

[6] Patel A, Shah J. Sensor-based activity recognition in the context of ambient assisted living systems: A review. Journal of Ambient Intelligence and Smart Environments. 2019;**11**(4):301-322

[7] Sulis E et al. An ambient assisted living architecture for hospital at home coupled with a process-oriented perspective. Journal of Ambient Intelligence and Humanized Computing. 2022;**2022**:1-19

[8] Rodrigues GN, et al. Dependability analysis in the ambient assisted living domain: An exploratory case study. Journal of Systems and Software. 2012;**85**(1):112-131

[9] Rashid J. Towards the development of an ubiquitous networked robot systems for ambient assisted living. In: 2010 IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing. IEEE; 2010. DOI: 10.1109/SUTC.2010.27

[10] Alosaimi W et al. Evaluating the impact of different symmetrical models of ambient assisted living systems. Symmetry. 2021;**13**(3):450

[11] Patel AD, Shah JH. Performance analysis of supervised machine learning algorithms to recognize human activity in ambient assisted living environment. In: 2019 IEEE 16th India Council International Conference (INDICON). IEEE; 2019. DOI: 10.1109/INDICON47234.2019.9030353

[12] Jaschinski C et al. Acceptance of technologies for aging in place: A conceptual model. Journal of Medical Internet Research. 2021;**23**(3):e22613

[13] Braunstein ML. Health Informatics on FHIR: How HL7's New API is Transforming Healthcare. Springer; 2018. ISBN: 978-3-319-93414-3

[14] Keroack MA. System integration: Managing complexity to advance health care value. NEJM Catalyst Innovations in Care Delivery. 2021;**2021**:2

[15] Sreenivasan M, Chacko AM. Interoperability issues in EHR systems: Research directions. Data Analytics in Biomedical Engineering and Healthcare. 2021;**2021**:13-28

[16] Diaz-Orueta U, Hopper L, Konstantinidis EJH. Shaping technologies for older adults with and without dementia: Reflections on ethics and

preferences. Health Informatics Journal. 2020;**26**(4):3215-3230

[17] Fonseca X. Location-Based Games for Social Interaction in Public Space. Delft; 2021. p. 318

[18] Fonseca X, Lukosch S, Brazier F. Fostering social interaction in playful cities. In: Interactivity, Game Creation, Design, Learning, and Innovation. Springer; 2018. pp. 286-295. DOI: 10.1007/978-3-030-06134-0_33

[19] Savvidis TP et al. Exergames for Parkinson's disease patients: How participatory design led to technology adaptation. In: Data, Informatics and Technology: An Inspiration for Improved Healthcare. IOS Press; 2018. pp. 78-81. DOI: 10.3233/978-1-61499-880-8-78

[20] Jin R, Pilozzi A, Huang X. Current cognition tests, potential virtual reality applications, and serious games in cognitive assessment and non-pharmacological therapy for neurocognitive disorders. Journal of Clinical Medicine. 2020;**9**(10):3287

[21] Rowlands DJH. What is digital health? And why does it matter. 2019

[22] Snowdon, A.J.H.I. and M.S. Society. Digital health: A framework for healthcare transformation. 2020

[23] Marques G, Bhoi AK, Hareesha K. IoT in Healthcare and Ambient Assisted Living. Springer; 2021. ISBN: 978-981-15-9897-5

[24] Tsai Y-T, Lin ZY. A survey on edge computing in bioinformatics and health informatics. In: 2020 IEEE International Conference on Bioinformatics and Biomedicine (BIBM). IEEE; 2020. DOI: 10.1109/BIBM49941.2020.9313103

[25] Calvaresi D, et al. Exploring the ambient assisted living domain: A systematic review. Journal of Ambient Intelligence and Humanized Computing. 2017;**8**(2):239-257

[26] Orlikowski WJJ. The duality of technology: Rethinking the concept of technology in organizations. Organization Science. 1992;**3**(3):398-427

[27] Choukou M, et al. Ambient assisted living technologies to support older adults' health and wellness: A systematic mapping review. European Review for Medical and Pharmacological Sciences. 2021;**25**(12):4289-4307

[28] Choukou MA et al. Evaluating the acceptance of ambient assisted living technology (AALT) in rehabilitation: A scoping review. International Journal of Medical Informatics. 2021;**2021**:104461

[29] Offermann-van Heek J, Schomakers E-M, Ziefle M. Bare necessities? How the need for care modulates the acceptance of ambient assisted living technologies. International Journal of Medical Informatics. 2019;**127**:147-156

[30] Offermann-van Heek J, Ziefle MJJ, a. technologies. They don't care about us! care personnel's perspectives on ambient assisted living technology usage: Scenario-based survey study. JMIR Rehabilitation and Assistive Technologies. 2018;**5**(2):e10424

[31] Konttila J et al. Healthcare professionals' competence in digitalisation: A systematic review. Journal of Clinical Nursing. 2019;**28**(5-6):745-761

[32] Gopal G et al. Digital transformation in healthcare–architectures of present and future information technologies. Clinical Chemistry and Laboratory Medicine (CCLM). 2019;**57**(3):328-335

[33] van Heek J, Himmel S, Ziefle M. Privacy, data security, and the acceptance of AAL-systems–a user-specific

perspective. In: International Conference on Human Aspects of IT for the Aged Population. Spring; 2017. DOI: 10.1007/978-3-319-58530-7_4

[34] Schwarz AF et al. Design features associated with user engagement in digital games for healthy lifestyle promotion in youth: A systematic review of qualitative and quantitative studies. Games for Health Journal. 2020;**9**(3):150-163

[35] Shin G et al. Wearable activity trackers, accuracy, adoption, acceptance and health impact: A systematic literature review. Journal of Biomedical Informatics. 2019;**93**:103153

[36] Fonseca, X., S. Lukosch, and F. Brazier. Secrets of the South: A Location-based Game for the Development of 21st Century Social Skills and Promotion of Social Interaction. in Proceedings of DELbA 2020 - Workshop on Designing and Facilitating Educational Location-based Applications (DELbA 2020) co-located with the Fifteenth European Conference on Technology Enhanced Learning (EC-TEL 2020), 2020. Heidelberg, Germany.

[37] Iancu I, Iancu B. Designing mobile technology for elderly. A theoretical overview. Technological Forecasting and Social Change. 2020;**155**:119977

[38] Lee L. Creativity and emotional attachment as a guide to factors of engagement for elderly interaction with technology. In: Proceedings of the 2019 on Creativity and Cognition. 2019. pp. 664-669. DOI: 10.1145/3325480.3326565

[39] Slingerland G et al. Location-based challenges for playful neighbourhood exploration. Behaviour & Information Technology. 2020;**2020**:433-451

[40] Hung L et al. Using touchscreen tablets to support social connections and reduce responsive behaviours among people with dementia in care settings: A scoping review. Dementia. 2021;**20**(3):1124-1143

[41] Fonseca X et al. Requirements for location-based games for social interaction. IEEE Transactions on Games. 2021;**14**(3):377-390

[42] Fonseca X et al. Designing for meaningful social interaction in digital serious games. Entertainment Computing. 2020;**36**(100385):1-23

[43] Przybylski AK, Orben A, Weinstein N. How much is too much? Examining the relationship between digital screen engagement and psychosocial functioning in a confirmatory cohort study. Journal of the American Academy of Child & Adolescent Psychiatry. 2020;**59**(9):1080-1088

[44] Bredl K, Bösche W. Serious Games and Virtual Worlds in Education, Professional Development, and Healthcare. IGI Global; 2013. DOI: 10.4018/978-1-4666-3673-6

[45] Sharifzadeh N et al. Health education serious games targeting health care providers, patients, and public health users: Scoping review. JMIR Serious Games. 2020;**8**(1):e13459

[46] Verschueren S, Buffel C, Vander Stichele G. Developing theory-driven, evidence-based serious games for health: Framework based on research community insights. JMIR Serious Games. 2019;**7**(2):e11565

[47] Haoran G, Bazakidi E, Zary N. Serious games in health professions education: Review of trends and learning efficacy. Yearbook of Medical Informatics. 2019;**28**(01):240-248

[48] Maheu-Cadotte M-A et al. Involvement of end users in the

development of serious games for health care professions education: Systematic descriptive review. JMIR Serious Games. 2021;**9**(3):e28650

[49] Brauner P, Holzinger A, Ziefle MJE. Ubiquitous Computing at its best: Serious exercise games for older adults in ambient assisted living environments-a technology acceptance perspective. European Union Digital Library. 2015;**1**(4):e3

[50] Brauner P, Ziefle MJJ. Serious motion-based exercise games for older adults: Evaluation of usability, performance, and pain mitigation. JMIR Serious Games. 2020;**8**(2)

[51] Wittland J, Brauner P, Ziefle M. Serious games for cognitive training in ambient assisted living environments–a technology acceptance perspective. In: IFIP Conference on Human-Computer Interaction. Spring; 2015. DOI: 10.1007/978-3-319-22701-6_34

[52] Göbel S et al. Serious games for health: Personalized exergames. In: Proceedings of the 18th ACM International Conference on Multimedia. 2010. DOI: 10.1145/1873951.1874316

[53] Garber CE et al. Quantity and quality of exercise for developing and maintaining cardiorespiratory, musculoskeletal, and neuromotor fitness in apparently healthy adults: Guidance for prescribing exercise. Medicine and Science in Sports and Exercise. 2011;**43**(7):1334-1359

[54] Morris CK et al. Nomogram based on metabolic equivalents and age for assessing aerobic exercise capacity in men. Journal of the American College of Cardiology. 1993;**22**(1):175-182

[55] Edwards TC et al. The metabolic equivalent of task score: A useful metric for comparing high-functioning hip arthroplasty patients. Bone & Joint Research. 2022;**11**(5):317-326

[56] Nooyi SC et al. Metabolic equivalent and its associated factors in a rural community of Karnataka, India. Cureus. 2019;**11**(6):1-9

[57] Holtermann A, Stamatakis E. Do All Daily Metabolic Equivalent Task Units (METs) Bring the Same Health Benefits? BMJ Publishing Group Ltd and British Association of Sport and Exercise Medicine; 2019. pp. 991-992. DOI: 10.1136/bjsports-2017-098693

[58] Bull FC et al. 2020 guidelines on physical activity and sedentary behaviour. British Journal of Sports Medicine. 2020;**54**(24):1451-1462

[59] O'Hara K, Glancy M, Robertshaw S. Understanding collective play in an urban screen game. In: Proceedings of the 2008 ACM Conference on Computer Supported Cooperative Work. 2008. DOI: 10.1145/1460563.1460576

[60] Awaisi KS et al. Leveraging IoT and fog computing in healthcare systems. IEEE Internet of Things Magazine. 2020;**3**(2):52-56

[61] Laroui M et al. Edge and fog computing for IoT: A survey on current research activities & future directions. Computer Communications. 2021;**180**:210-231

[62] Fonseca X et al. Requirements and Game Ideas for Social Interaction in Mobile Outdoor Games. In: CHI PLAY '17 Extended Abstracts, Publication of the Annual Symposium on Computer-Human Interaction in Play. 2017. pp. 331-337. DOI: 10.1145/3130859.3131304

Edited by Branislav Sobota and Emília Pietriková

Computer games have their basis in mainframe computers from the 1960s. Their spread among people came after the arrival of small (8-bit) home computers. In the beginning, their development was influenced by both the technical and program resources of these computers. These games usually form the first level of communication between a human and a computer. This situation is particularly striking for games on modern computers, game consoles, or smartphones. This book discusses modern approaches, procedures, and algorithms as well as devices in the field of computer game development, with a focus on both the influence of computer science on the development of computer games and the impact of computer games on the development of computer science. Among other things, the book explores virtual reality technologies and visualization, game cores (engines) and platforms, genres, formalizations, correct object-procedural graphic design, trends in interfaces and computer–player interaction, and the development of computer games.

Published in London, UK

IntechOpen

ISBN 978-1-83769-734-2