

VIDEO JOGOS 2016

UNIVERSIDADE DA BEIRA INTERIOR
24 E 25 NOVEMBRO

VIDEOJOGOS

UBI, 24 E 25 DE NOVEMBRO DE 2016

Atas da Videojogos 2016 - 9ª Conferência de Ciências e Artes dos Videojogos

24 e 25 de novembro de 2016, Universidade da Beira Interior, Covilhã

Editores

Frutuoso Silva e Pedro Santos

ISBN: 978-989-20-7148-0

Edição Electrónica, 107 páginas

Ano: 2016

Capa e contracapa

Fotografias da Serra da Estrela tiradas por Frutuoso Silva

VIDEOJOGOS

UBI, 24 E 25 DE NOVEMBRO DE 2016

Prefácio

Apresentamos neste volume as atas da nona edição da conferência anual Videojogos, a Videojogos 2016 - Conferência de Ciências e Artes dos Videojogos. A Videojogos 2016 foi organizada este ano pela Universidade da Beira Interior. Decorreu nos dias 24 e 25 de novembro na Covilhã e, como tem sido habitual desde 2009, em colaboração com a Sociedade Portuguesa de Ciências dos Videojogos. Este ano, a conferência foi co-localizada com o Encontro Português de Computação Gráfica e Interação (EPCGI'2016).

A conferência seguiu a tradição de divulgar e estimular o trabalho de investigação e desenvolvimento de videojogos no espaço lusófono. O evento manteve a sua essência multidisciplinar com a apresentação de trabalhos cobrindo um vasto leque temas que foram desde a situação do setor de criação de videojogos em Portugal até à escolha e composição musical para videojogos, passando por motores de jogo e técnicas de Inteligência Artificial.

Tentámos este ano dar particular ênfase aos contactos com as empresas que criam videojogos ou componentes, que se traduziu em dois dos oradores convidados, Rui Casais, CEO da Funcom, uma empresa Norueguesa especializada em jogos MMO (Massively Multiplayer Online), e Francisco Rebelo de Andrade, co-fundador da Sceelix, uma empresa Portuguesa especializada em Middleware. Ambos focaram as suas apresentações no empreendedorismo na área de software e videojogos. Contámos ainda com a apresentação convidada de Alan Dix, conhecido professor e investigador na área da Interação Pessoa-Máquina na University of Birmingham.

A conferência foi encerrada com a entrega dos certificados de melhor artigo e melhor demo apresentados na conferência.

No total, foram submetidas à conferência 22 artigos e 4 demos, tendo sido seleccionados 6 Full Papers, 7 Work in Progress e 3 Demos para apresentação na conferência e nestas atas. A conferência contou com 25 participantes inscritos dos quais 11 estudantes. Foi infelizmente afectada pela situação política no Brasil, que impediu que uma série de colegas Brasileiros pudessem tomar parte.

Por fim, gostaríamos de agradecer aos autores, aos oradores convidados, aos membros da Comissão Científica, aos membros da Comissão Organizadora, aos Patrocinadores e à Universidade da Beira Interior pelas instalações e suporte que permitiram a realização deste evento. Um agradecimento especial à Sociedade Portuguesa de Ciências dos Videojogos, cuja dedicação e esforço tem permitido a realização da Videojogos ao longo dos anos.

Dezembro de 2016

Frutuoso Silva
Pedro Santos

VIDEOJOGOS

UBI, 24 E 25 DE NOVEMBRO DE 2016

Organização, Apoios e Patrocinadores

Presidência da conferência

Pedro Santos (IST)
Frutuoso Silva (UBI)

Comissão organizadora

Frutuoso Silva (UBI)
Herlander Elias (UBI)
Gonçalo Fonseca (UBI)
Inês Lopes (UBI)
Nuno Roxo (UBI)

Comissão Científica

Pedro Santos (IST e INESC-ID) - Chair
Ana Sofia Almeida (Tapestry)
Ana Veloso (Univ. Aveiro)
André Neves (UFPE)
A. Augusto Sousa (FEUP e INESC Porto)
António Coelho (FEUP)
António Ramires (Univ. Minho)
Carlos Martinho (IST e INESC-ID)
Cátia Ferreira (UCP)
Danielle Rousy Dias da Silva (UFPE, Brasil)
Diana Adamatti (FURG, Brasil)
Esteban Clua (ICUFF/UFF Media Lab, Brasil)
Eva Oliveira (IPCA)
Filipe Luz (ULHT / MovLab)
Filipe Penicheiro (Univ. Coimbra)
Frutuoso Silva (UBI)
José Azevedo (FLUP)
João Dias (IST e INESC-ID)
João Madeiras Pereira (IST-UTL)
Leonel Morgado (UAb)

Licínio Roque (Univ. Coimbra)
Lucia Amante (UAb)
Luís Filipe B. Teixeira (CIES-IUL)
Luís Pereira (Univ. Minho)
Lynn Alves (UNEB)
Mário Vairinhos (Univ. Aveiro)
Nelson Zagalo (Univ. Minho)
Nuno Rodrigues (IPCA)
Óscar Mealha (Univ. Aveiro)
Patrícia Gouveia (FBAUL)
Patrícia Tedesco (UFPE, Brasil)
Paulo Dias (Univ. Aveiro)
Paulo Quaresma (Univ. Évora)
Pedro Cardoso (FBAUP)
Roger Tavares (UFRN, Brasil)
Rui Prada (IST e INESC-ID)
Samuel Mascarenhas (IST e INESC-ID)
Teresa Romão (FCT-UNL)
Valter Alves (IPViseu / Univ Coimbra)

Apoios:



Patrocinadores:



VIDEOJOGOS

UBI, 24 E 25 DE NOVEMBRO DE 2016

Índice

Prefácio	ii
Organização, Apoios e Patrocinadores	iii
Palestras Convidadas	vi
Videojogos – Como sobreviver e crescer numa indústria em constante mudança	vi
<i>Rui Casais</i>	
Sceelix procedural engine: da tecnologia ao produto	vi
<i>Francisco Rebelo de Andrade</i>	
Open Data Islands and Communities	vi
<i>Alan Dix</i>	
The video game industry in Portugal	1
<i>Pedro A. Santos, Patrícia Romeiro, Flávio Nunes, Paul Hollins and Ruben Riestra</i>	
Gaming and VR Technologies, Powers and Discontents	11
<i>Patrícia Gouveia</i>	
MiniPool: Real-time artificial player for an 8-Ball video game	21
<i>David Silva and Rui Prada</i>	
“So medieval like, so gentle, so perfect”: as categorias musicais da banda-sonora do videojogo The Elder Scrolls IV: Oblivion	31
<i>Joana Freitas</i>	
Towards a Procedurally Generated Experience: A Structural Analysis of Quests	41
<i>António Machado, Pedro Santos and João Dias</i>	
A representação subjetiva dos signos culinário e a polícia no jogo Max Payne 3. Uma análise semiótica da representação imagética em jogos de video game	51
<i>Carlos William F. de Lima</i>	
JOT: A Modular Multi-purpose Minimalistic Massively Multiplayer Online Game Engine	57
<i>Gonçalo Amador and Abel Gomes</i>	
Arquitecturas de rede em jogos multi jogador no Unity	63
<i>Nuno Carapito e Frutuoso Silva</i>	
Análise heurística de jogos digitais multijogador em rede: O caso do “Curve Fever 2”	71
<i>Cláudio Duarte e Ana Isabel Veloso</i>	

VIDEOJOGOS

UBI, 24 E 25 DE NOVEMBRO DE 2016

CODing theGAME: projeto de um videodocumentário sobre a cena de produção de jogos digitais em Portugal	79
<i>Rogério Tavares e Licínio Roque</i>	
Exploração do High Fidelity para desenvolvimento de jogos em mundos virtuais imersivos multiutilizador	85
<i>José Martins, Leonel Morgado e Vítor Cardoso</i>	
Fan culture, comunidades e mods: ensaio sobre a dimensão do multiverso cultural dos games na internet	91
<i>Andre Pequeno Dos Santos e José Manuel Azevedo</i>	
Desenvolvimento de um Jogo para o ensino de Física com Design Centrado no Usuário	97
<i>Jessica Leite Pituba e Ricardo Nakamura</i>	
Jogos sérios digitais para a promoção do património natural – o caso de um jogo de exploração de um parque na busca dos seus animais	105
<i>Liliana Santos, Daniel Pereira, Rui Nóbrega, Pedro Beça e António Coelho</i>	
Inmension – Vem Descobrir a Realidade Aumentada	106
<i>Gregory Caldeira, José Dias, Ana Isabel Veloso, Hélder Caixinha e Pedro Amado</i>	
Forgotten Shaft – Demo	107
<i>Bruno Lima, Eduardo Gualdino, Rennan Raffaele e Frutuoso Silva</i>	

VIDEOJOGOS

UBI, 24 E 25 DE NOVEMBRO DE 2016

Palestras Convidadas

Videojogos – Como sobreviver e crescer numa indústria em constante mudança

Rui Casais, CEO da Funcom

A indústria de videogames tem-se caracterizado por uma rápida evolução e pela forma imparável com que persegue novas tecnologias. Estas características geram inúmeras oportunidades de negócio mas em contrapartida também são responsáveis por criarem um ambiente difícil de controlar e prever.

A Funcom é um exemplo deste fluxo constante nesta indústria, tendo crescido e contraído várias vezes durante os 12 anos em que Rui Casais lá está a trabalhar, e é com base nesta experiência que serão dados exemplos de dicas e conhecimentos adquiridos para não apenas sobreviver, mas também crescer, neste mercado bem competitivo.

Sceelix procedural engine: da tecnologia ao produto

Francisco Rebelo de Andrade, Co-fundador da Sceelix

Sceelix, Create and Manage your 3D Scenes Smartly. The 3D Scenes Procedural Engine

Open Data Islands and Communities

Alan Dix, University of Birmingham

How do we make digital technology serve those at physical and social margins of society?

Digital technology, not least the Internet, has transformed many aspects of our lives. Crucially, in many countries access to digital technology has become an essential part of the nature of modern citizenry for commercial services; for access to government, and for participation in democratic processes, for example much of the UK Brexit and US Presidential campaigns were fought on Facebook.

However, the ability to take advantage of digital technology is not uniform, those at the margins typically have disproportionately poor access, both in terms of physical connectivity and skills. There is a danger that digital technology can deepen the existing divides in our world.

In this talk I will look at these issues and most importantly ways we can, as researchers and practitioners, seek to create technologies that serve all communities. I will focus particularly on open data, how we can devise ways to make it more easily found, accessed, and visualised by small communities at the edges, and moreover how they can become active creators of information: producers not merely subjects of data.

I will draw on experience in a number of projects on the small Scottish island of Tiree and also my 1500 kilometre walk around the edges of Wales.

The video game industry in Portugal

Pedro A. Santos
INESC-ID / Instituto Superior
Técnico
Universidade de Lisboa
pedro.santos@tecnico.ulisboa.pt

Patrícia Romeiro
CEGOT
Universidade do Porto
patriciaromeiro@yahoo.com

Flávio Nunes
CEGOT
Universidade do Minho
flavionunes@geografia.uminho.pt

Paul Hollins
The University of Bolton
paulanthonyhollins@googlemail

Ruben Riestra
Inmark Europa S.A.
ruben.riestra@grupoinmark.com

ABSTRACT

Despite the impressive growth of the video game industry in Europe and a growing interest emerging in that industrial sector (e.g. from the European Commission), there is still a knowledge deficiency in respect of its characteristics and regional impact. By mapping the most relevant active agents participating in this economic activity in Portugal, this paper explores the characteristics the industry communities, types of games developed, tools used, marketing and sales channels and economic impact. This research is based on data collected in Portugal through a survey of 70 economic agents, under the auspices of the *First Atlas of video games Industry in Portugal* research project. The paper concludes with an elaboration of the opportunities and challenges associated with the industry and the potential role of public and institutional policies in supporting its development and sustainability.

Author Keywords

Video Game Industry, Creative Industries, Public Policies, Portugal

INTRODUCTION

The video game industry is currently exhibiting greater growth potential than many others of the creative industries. In 2015 worldwide revenues were estimated to be between \$61 billion¹ and \$91.5 billion², representing an 8% increase in comparison with 2014. This level of growth is predicted to continue over the next few years. In many European countries, the video game industry represents an important part of the GDP, greater than other creative industries such as Music or Film. This

¹<http://www.cnn.com/2016/01/26/digital-gaming-sales-hit-record-61-billion-in-2015-report.html>

²<https://newzoo.com/insights/articles/us-and-china-take-half-of-113bn-games-market-in-2018/>

occurs in the United Kingdom, Sweden and Finland, where this sector contributes with more than \$1 billion to the GDP of these three countries^{3 4 5}.

Video games with good production values can now be made relatively inexpensively by small teams using tools such as Unity and Unreal. Games can be sold and distributed electronically in large quantities, using Apple or Google stores in the case of mobile apps, or Steam in case of PC games. Hardware and Console manufacturers have also been lowering the barriers to entry for small and independent developers. Since 2008 these factors have led to an explosion of game production by small teams. There are numerous examples. Minecraft started in 2009 as an independent project by Swedish programmer Markus Alexej Persson and the company he founded was sold to Microsoft in 2014 for \$2.5 billion⁶. Supercell is a mobile game development company founded in June 2010 in Helsinki, Finland with \$250 000 initial investment by its founders. The company saw its valuation double to \$10.2 billion in the last 12 months after its games Hay Day, Clash of Clans and Clash Royale attained worldwide success⁷.

In parallel, games are being used for multiple purposes, for entertainment and increasingly for learning, and many European businesses are taking this opportunity [1, 2]. One can ask then what is the impact of the current *low barrier to entry - crowded environment - high prizes for success* environment on the Portuguese industry?

The Portuguese video game industry history can be divided into three distinct phases [9]. The first phase, starting in the 1980s was associated with the success of the first personal computers (PC) and was driven by young amateur programmers. The second phase, of the mid 1990s, was associated with the establishment of the Internet. The Internet provided a strong impulse to the development of video games in Portugal

³<http://www.tiga.org/news/the-uk-video-game-development-sector-is-back-on-track>

⁴<http://www.swedishgamesindustry.com/news/2016/9/23/swedish-game-exports-closing-in-on-truck-sales.aspx>

⁵<https://www.tekes.fi/en/whats-going-on/news-from-tekes/finnish-game-industry-grew-into-a-two-billion-euro-business/>

⁶<https://en.wikipedia.org/wiki/Mojang>

⁷<http://www.dfcint.com/dossier/softbank-sells-supercell-for-8-6b/>

by facilitating information access via forums and online tutorials. Finally, a more recent third phase witnessed the formation of game development studios. In the last few years the higher education (HE) sector, identified the potential student interest and commenced offering specific degrees in the various disciplines of video game development.

This paper provides an insight in to the current state of the Portuguese video game industry in 2016, in presenting preliminary results from the project *First Atlas of video games Industry in Portugal*, developed by a team from INESC-ID and CEGOT, a study integrated in the European Project RAGE (Realising an Applied Gaming Eco-System). This project has the objective of strengthening the European video game industry, namely the applied games sector. This study represents an initial attempt to map and systematically characterise the Portuguese video game industry.

The Portuguese video game industry has thus far not been seriously studied, due largely to its emergent state and informality. There has been some work focused on the historical perspective (see [9]) and on some specific regions within Portugal (see [6]). There has been little analysis on the economic activity generated by video game development (see [7]). In this study the authors contribution overcomes this lack of information and provides new insights into the dimension and characteristics of this still immature and emergent industry and provides the information needed to support the strategies and policies required to the strengthen and ensure sustainability of the industry in Portugal.

The first section of this paper describes and discusses the methodology used in the study. This is followed by a section presenting the results and analysis, organised into five key areas: businesses, employment, products, economic situation and institutional support. The final section speculates on the future prospects of the industry as seen by the economic agents and actions the authors suggest should be taken to support the industry.

METHODOLOGY

In order to prepare a characterisation of the video game industry in Portugal we applied a mixed methodology that referenced a wide range of information sources : websites and trade social networks; general and specialised media; aggregator or brokerage platforms; several publications (e.g. [4, 5, 9]); attendance lists at industry events; to supplement the personal contacts of the project team. Triangulation of these sources resulted in the creation of a database that aggregates the businesses and creators engaged in the industry, as well as another database that identifies video games produced and marketed in Portugal to date. The task was hampered by several factors, including the wide dispersal of information on the sector, high business mortality rate, the intense mobility of workers and contradictory data.

This was the starting point for the collection of primary data, by conducting a survey directed toward businesses and individual entrepreneurs (hereinafter referred to as Businesses) and the self-employed and freelancers from this sector (hereinafter called Creators). The survey was available for completion

online between December 2015 and February 2016 and was directly publicised to the businesses and creators (using e.g., email, social networking, phone) or through aggregator platforms and intermediary organisations (e.g., SPCV⁸, university research centres, incubators and business centres). A total of 38 businesses responded to the survey (representing 20.7% of the 183 businesses inventoried in the database – some are no longer active – and 49.4% of the 77 businesses that we confirmed were still operating) and 32 independent creators for a total of 70 agents involved in the video game industry in Portugal.

In order to collect detailed information on the operation of the video game industry in Portugal, we also carried out 12 in-depth interviews to targeted businesses, selected to cover a range in terms of volume of business, type of produced video games (applied or entertainment) and geographical location. For this paper the data was particularly useful in order to provide context for the survey. It will also be subject to in-depth treatment in the context of other outputs to be developed in the near future by the team.

This paper presents the preliminary results of the study. A more detailed analysis and the complete results are currently being compiled as part of a report to be published by SPCV and Rage Project[8]. It should be noted that it was not the aim of this (first) study to cover the entire video game economic footprint in Portugal, such a project would include the import, marketing and sales of video games, together with game's journalism, video content producers and research activities. We opted, given the challenges and limited resources to cover only the game development industry. .

A broader research study of the video game sector in Portugal should be a priority for subsequent study.

RESULTS AND ANALYSIS

The authors will present and analyse next the results obtained from the survey and interviews to businesses and creators, characterising the state of the industry in five key areas: businesses and creators, employment, products, economic situation and institutional support.

Businesses and Creators

Respondent businesses and creators have emerged from 2001, with between 1 and 3 businesses and creators commencing activity per year through to 2011, with the one exception of 2005 when no new Businesses appeared. This trend was succeeded by a new phase in which the sector's dynamism has strengthened considerably, with the annual appearance of 10 or more businesses and creators from 2012. The most active year was 2013 when 21 new agents started activity (11 businesses and 10 creators). There appears to be a decrease in new activity over the last year (2015), but that observation could be due to new ventures that started recently being operating below the radar coverage, until results are visible.

It is thus a very young industry, evidenced not only by the fact the oldest company emerged only in 2001, but also by the fact that almost two thirds of businesses (65%) have had 3 or

⁸Sociedade Portuguesa de Ciências dos Videojogos

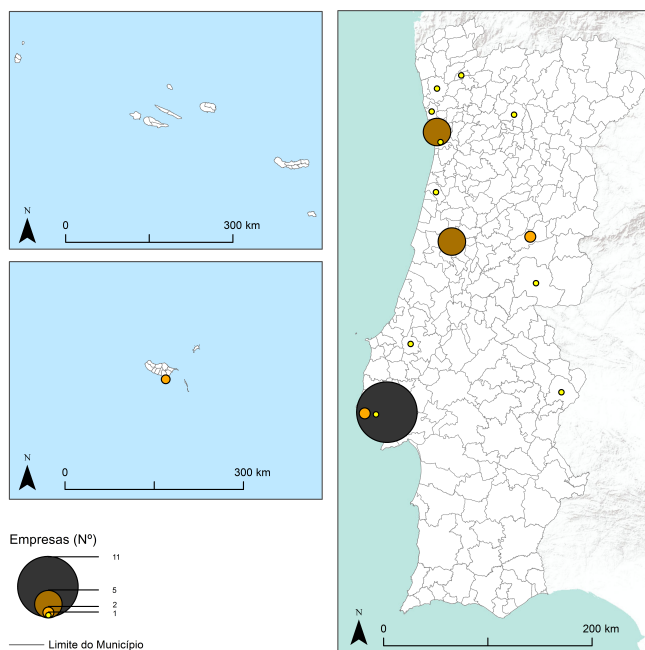


Figure 1. Geographical distribution of game development businesses in Portugal, 2016. Source: Survey to Businesses and Creators (January 2016)

fewer years of activity. The information from the interviews suggests that this recent dynamism is mainly the result of the (re)emergence of game platforms – PC , mobile, web – and easy access to more intuitive tools for creating and editing video games.

The analysis of the geographical distribution of the surveyed businesses (see Figure 1) indicates that this is an economic activity still not distributed evenly in Portugal and concentrated in some cities. Most municipalities have no businesses or creators, as they are concentrated on only 16 of the 308 Portuguese municipalities

Only three municipalities recorded more than 3 businesses on its territory, Lisbon having 11 units, compared to Porto and Coimbra that each have 5 businesses. If we to these also join the creators, individually dedicated to video game development in Portugal, the role of the capital is even more remarkable, with a total of 24 businesses/creators, this is followed by Porto with 8 and Coimbra with 7. It must be also mentioned that the municipality of Oeiras hosts the largest Portuguese studio, with over a hundred employees.

Answers to questions on the factors influencing location showed that businesses and creators do not attach the same order of importance the factors that in some way influence their location. For businesses the most relevant factor is their proximity to other complementary creative industries, the intersection of learning that such proximity provides. This is followed by, in order of importance, the proximity to other businesses or creators who develop video games and then the ease of interaction with universities, the inference being that these are the source of preferential recruitment for the building

Activity	Businesses		Creators	
	Number	%	Number	%
Original Game Dev.	37	100.0	28	87.5
Design / Prototyping	10	27.0	19	59.4
Publishing	9	24.3	4	12.5
Artistic Creation	8	21.6	13	40.6
Subcontractor	7	19.0	22	69.0
Distribution/Exploitation	4	10.8	3	9.4
Porting	3	8.1	2	6.3
Middleware Dev.	2	5.4	0	0.0
Research	0	0.0	1	3.1

Table 1. Main activities. Source: Survey to Businesses and Creators (January 2016)

of work teams. In turn, creators attach more importance to the existence in the vicinity of other businesses or creators who develop video games, as a result of the working links that often establish with these agents. The second locative factor most valued by creators is the proximity to other creative industries. The proximity to research centres is assumed as the third most important factor that determines the location of where the creators operate.

In terms of activities, the majority of businesses are dedicated exclusively to the development of video games. In fact , less than 1/3 of these accumulates this activity with other services not directly related to the creation of video games. However, the same situation does not occur with creators, as most of them accumulate the development of digital games with other activities. Of the businesses that are not dedicated exclusively to the development of video games (32% of all surveyed businesses) most are involved in activities related to programming, as well as web design and consultancy. The creators who are not exclusively dedicated to the development of video games (56% of respondents) join that activity with involvement in activities related to programming (similar to what happens with the businesses), or other digital activities such as illustration, multimedia animation or training.

Analysing the main activities performed in the development of video games one can detect, in the case of businesses, a tendency for a greater dispersion of their expertise. All businesses indicate the creation of original games as a main activity. Then, among the other activities stand out design and prototyping, publishing and artistic creation (sound, 3D modeling , illustration, ...). Creators tend to demonstrate more a concentrated expertise profile focused on two distinct activities. These are design and prototyping, activity performed for about 60% of creators, and artistic creation that was mentioned by about 40%. Video game ports for other platforms and the development of middleware (to integrate in the pipeline from other businesses) are done only by a residual percentage, both in the case of surveyed businesses and creators (see Table 1).

Employment

The surveyed businesses reported 272 full-time and 48 part-time employees, and 55 freelancers, which corresponds to a total of 375 workers. Thus, it is estimated that the total num-

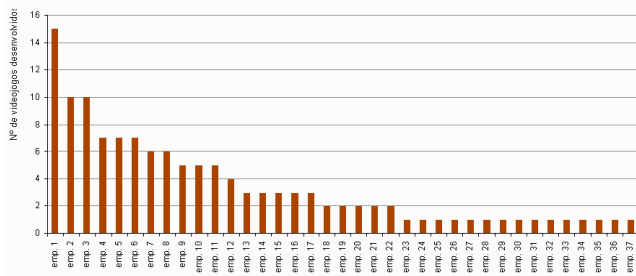


Figure 2. Games developed by businesses in Portugal. Source: Survey to Businesses and Creators (January 2016)

ber of direct jobs generated by the industry in Portugal is in the range of 658-1204 employees. The values that define the boundaries of this range were extrapolated based on a projection calculated from the sample used in this study, which represents 25% of the 154 businesses that are in the project database and about 50% of businesses confirmed to be effectively in operation –77 businesses⁹. These calculations do not consider the creators identified in the database, in order to avoid double counting of workers (many of whom are referred to by the businesses as a freelancer workers).

Although the average number of employees per company being 10.1, infers an industrial base composed of small businesses, this does not correspond to the reality of this sector. If we subtract the 120 employees that are allocated to a single company (based in Lisbon Metropolitan Area) from the total of 375 workers who collaborate with these businesses, we get an average of 7 workers per company (or only 4 workers per company considering full-time employees). This is similar to other sectors with higher levels of expertise and which are technology-intensive and knowledge-intensive, this is an ecosystem consisting of micro enterprises.

Portuguese Video Games

The businesses surveyed were responsible for the development of 127 video games. Of this total, 110 (92%) were sold on the international market. This stems from the facility that currently exists in publishing games for mobile platforms or PC through the App stores and Steam as well as the small size of the domestic market.

On average each company produced approximately 3 games. This figure however hides a great disparity of situations in the number of video games produced. One company has produced 15 of these games and there are 15 that were responsible for the development of a single one (Figure 2).

Genre, target audience, tools

There is a wide variety in the genre of video games developed either by businesses or by the creators. The genres most represented in the portfolio of Portuguese businesses and creators are arcade, action, puzzle and adventure games. At least 30% of businesses and/or creators claim to have developed these

⁹To calculate these estimates we have not considered the 120 employees working in one company that has responded to the survey, because that number represents a singular reality.

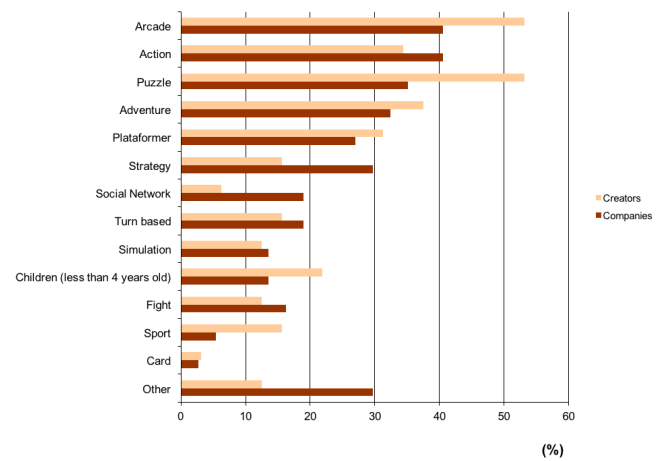


Figure 3. Percentage of businesses and creators working by genre. Source: Survey to Businesses and Creators (January 2016)

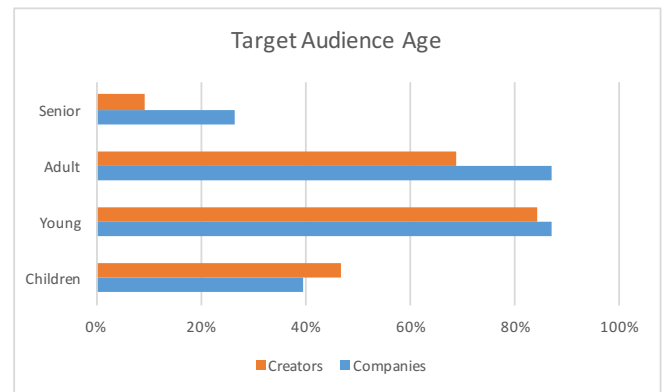


Figure 4. Target audience the agents produce games for. Source: Survey to Businesses and Creators (January 2016)

types of video games. It should be noted that over 50% of the creators said they had worked in either Arcade games or Action games (Figure 3).

Video games developed by businesses and creators share a similar target audience (Figure 4) Video games developed in Portugal are intended primarily for young people and adults, following by children and, finally, the senior public.

Both businesses and creators develop games primarily for Mobile and PC, with about a third of businesses and 15% of creators having worked or currently working on console games. Also about a third of the businesses and creators worked or are working in video games designed to be played online (Figure 5).

Regarding development tools, Unity3D is currently the dominant development platform among the Portuguese businesses and creators. More than 70% say they use it. In a distant second place comes the Unreal Engine used by about 13% of businesses and creators, and in third place Game Maker Studio, used by 8% to 9% of cases (Table 2). Note also that Co-

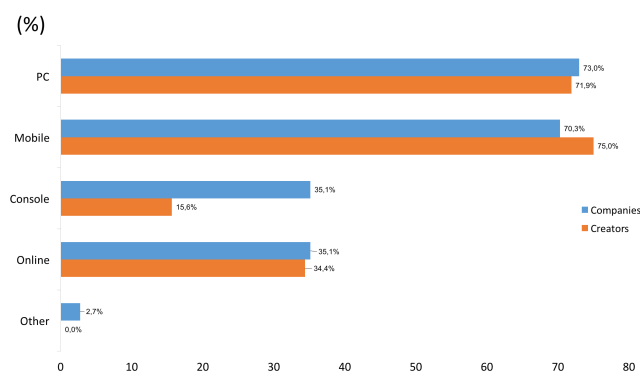


Figure 5. Target platforms of Portuguese games. Source: Survey to Businesses and Creators (January 2016)

Tool / Platform	Businesses	Creators
Unity3D	73.0	75.0
Unreal Engine	13.5	12.5
Game Maker Studio	8.1	9.4
HTML 5	8.1	0.0
Cocos 2D	5.4	0.0
Haxe Family	5.4	0.0
XNA	5.4	0.0
Own Engine	5.4	0.0
Flash	0.0	9.4
Libgdx	0.0	9.4

Table 2. Main development tools and platforms. Source: Survey to Businesses and Creators (January 2016)

cos2D is a main tool and platform used by the largest company considered in this study.

Marketing and distribution

The main channels used for advertising video games developed in Portugal are social networks and the businesses' own websites. For about 60% of the businesses, but only 30% of creators, the specialized press is also a channel used. Youtubers or an owned YouTube channel, as well as the video game industry events are also channels with expression which are used by about half of the businesses and between 20% and 40% of the creators. About 35% of businesses and creators resort to advertising. For cross advertising one needs to have already a successful portfolio and so this channel is less used (Figure 6).

In terms of distribution channels and sales, this is dominated by the Google and Apple stores, with more than 60% of businesses declaring their use. Creators use less the Apple App Store (less than 45% compared to over 70% for Google Play), probably derived from its annual cost. The Microsoft service, Windows Store, has just over 30% of businesses and 20% of creators as users (Figure 7).

Steam is used by 40% of Enterprises and just over 10% of creators. The reason for this difference is not clear, since

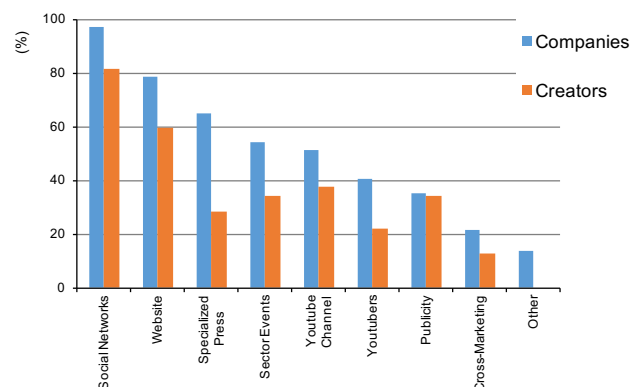


Figure 6. Marketing of Portuguese games. Source: Survey to Businesses and Creators (January 2016)

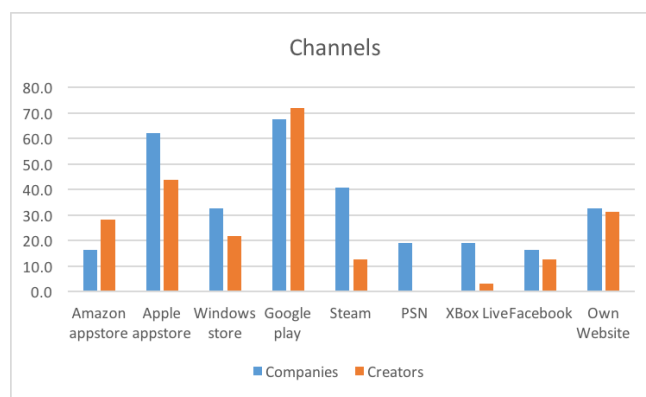


Figure 7. Distribution Channels. Source: Survey to Businesses and Creators (January 2016)

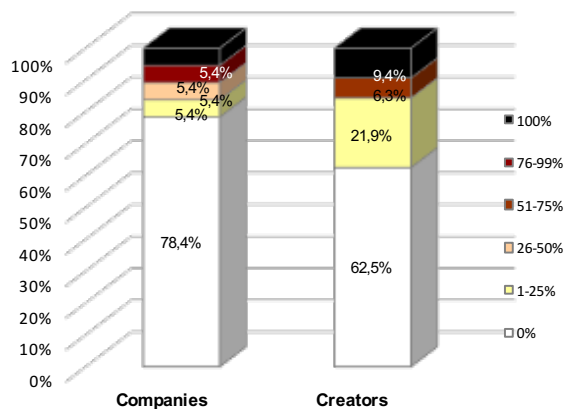


Figure 8. Percentage of income derived from applied games. Source: Survey to Businesses and Creators (January 2016)

both declare to develop PC games. This could be due to the fact that many of the creators working on PC games do not publish them directly, this being an activity that requires more resources. To publish a game on Steam, for example, it is currently required to go through a greenlight quality process, which requires additional effort and some investment. About 20% of businesses distribute their games on digital channels connected to game consoles such as Xbox Live or PSN. Their own website is used by about 30% of businesses and creators, Other channels, such as Facebook or physical distribution are not currently significant.

Applied Games in Portugal

Only about 11% of businesses and 16% of creators declare that applied games represent most of their income, while another 10% of the businesses and 20% of creators declare some activity in this type of games (see Figure 8).

Regarding the sectors of application of the games produced, 60% of businesses and 75% of creators working on Applied Games indicate education and formation. There is also some presence in the areas of information/communication, health, logistics, architecture and construction, security, and insurance (see Figure 9).

Economic Situation

For the development of video games in Portugal and in terms of funding there is a heavy reliance on equity, with less than 50% of businesses and creators referring recourse to alternative sources of funding. This is a signal of the lack of connection between the videogame production industry players in Portugal and possible investors (see Figure 10).

It should be noted the lack of venture capital expression in game development in Portugal, as well as the limited use still made by Portuguese businesses and creators of alternative funding sources such as crowdfunding .

About 95% of businesses and 85% of creators assess access to credit as difficult or very difficult. This difficulty experienced in accessing credit helps explain the weak presence of the

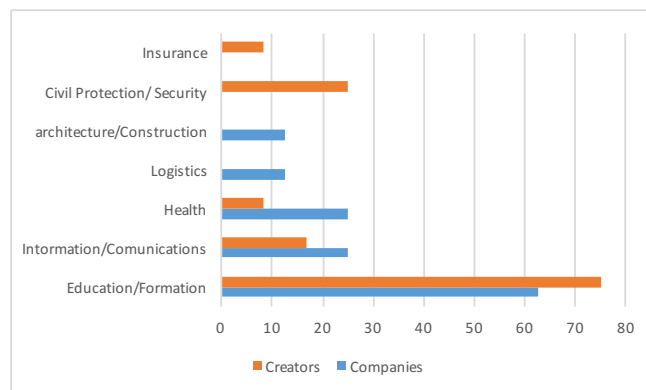


Figure 9. Percentage of income derived from applied games. Source: Survey to Businesses and Creators (January 2016)

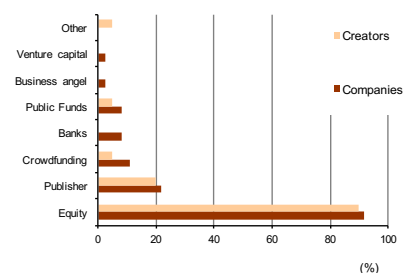


Figure 10. Origin of the financing for game production in Portugal. Source: Survey to Businesses and Creators (January 2016)

traditional sources of financing for the development of video games in Portugal (see Figure 11).

Regarding the sources of revenue associated with the video games produced in Portugal, businesses mention that the most common source is the sale of the game itself (65%). About half of these agents also report sales within the game (micro-transactions) and just over 40% refer to advertising or sponsorship. This final source of income is the most common among creators, with over 50% mentioning it. Other sources of revenue such as subscription, crowdfunding or early access, services or merchandising sales are always referred to by less than 10% of respondents (see Figure 12).

Only about 20% of businesses reported having revenues exceeding 50,000 Euros annually, with one third of businesses saying they had income lower than 10 000 Euros per year, which would not make them sustainable as sources of employment (Figure 13). As for the creators, 56% said they had annual revenues related to video games of less than 2,000 Euros.

This infers that many businesses and creators are still in an early investment phase, without their games generating revenue that enables a sustainable activity. Note that three businesses declared turnovers of between 250,000 and 500,000 Euros. It should also be noted that several businesses have declined to reveal their revenues, including the company that employs the most people (in our sample).

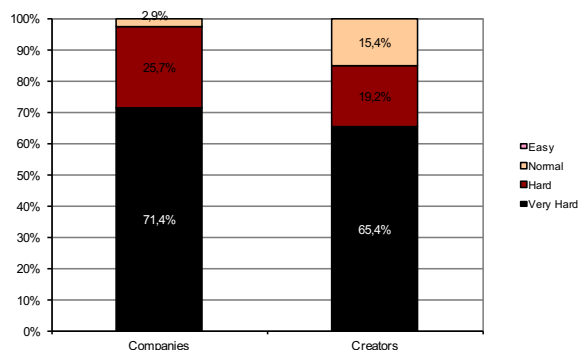


Figure 11. How easy is to access credit. Source: Survey to Businesses and Creators (January 2016)

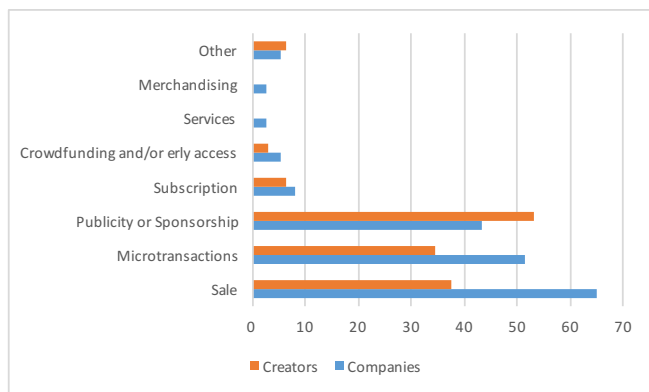


Figure 12. Revenue streams. Source: Survey to Businesses and Creators (January 2016)

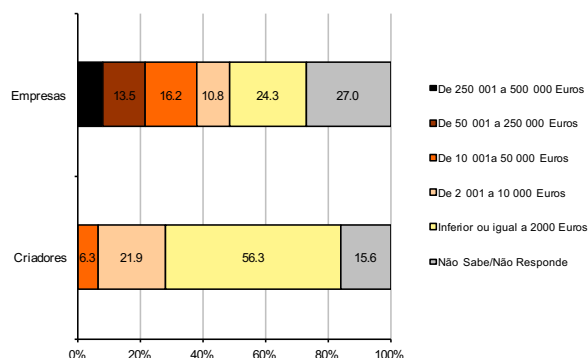


Figure 13. Annual Revenue. Source: Survey to Businesses and Creators (January 2016)

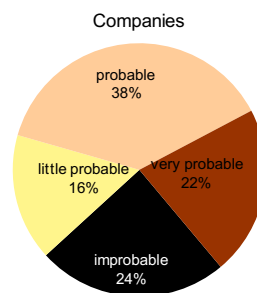


Figure 14. Future support application. Source: Survey to Businesses and Creators (January 2016)

Despite the high percentage of businesses that claims to have low revenues, over 70% of businesses consider their financial situation regular or favorable, which corroborates our assertion that these businesses are in their investment phase. On the other hand, on the creators side, the financial situation is a cause for concern in 75% of cases.

To determine the overall turnover of the videogame industry in Portugal, we consider the average value of each class considered in this question, and just for the sample businesses that responded we obtain the approximate annual total amount of 2,100,000 Euros. Thus, considering the employment data, we estimate the total turnover generated by the sector's businesses in Portugal to be in a range of between 6 and 12 million Euros.

Institutional Support

It is not frequent the recourse to public support for the development of video games in Portugal. In fact, 94.6% of businesses and 62.5% of creators have not benefited from any support of this kind for the development of their activity. A small number of businesses benefited from support of the programs 'Compete'- QREN and MEDIA or MEDIA Mundus. In the case of the 37.5% of creators who were recipients of public support, we highlight the MEDIA and MEDIA Mundus programs, support from University institutions and the support provided by local authorities.

The public support that benefited businesses and creators had as a purpose the development of video games. Specifically, this was the sole purpose identified by the businesses (66.7%), while the creators received support not only for the development of video games (50%), but also for the establishment of partnerships and networks (25%), the acquisition new technologies (15%) and , to a lesser extent, access to new markets (5%).

The future application for national or European support programs is mostly likely to support businesses. In fact, most of these (60%) consider as 'probable' or 'very probable' that this will happen, in contrast to the 31% probability in the case of creators. Many of these creators (47%) consider 'Improbable' future application for public support (see Figure 14).

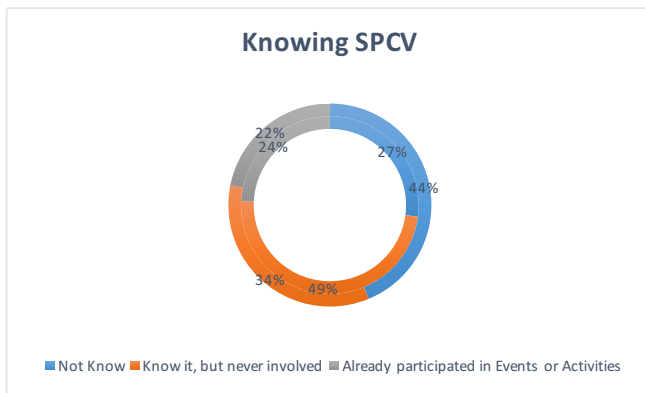


Figure 15. Relation with SPCV; outer circle represents the creators. Source: Survey to Businesses and Creators (January 2016)

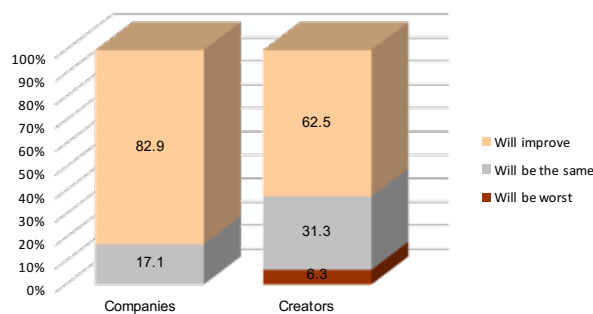


Figure 16. Confidence in the future. Source: Survey to Businesses and Creators (January 2016)

A final question in the institutional support section of the survey regarded the ‘Sociedade Portuguesa de Ciências dos Videojogos’ (SPCV). This is a non-profit association that aims to promote and develop the science of video games in Portugal. In the absence of a proper industry body, SPCV, despite having academic roots, has tried to promote dynamics in video game industry community.

The answers to the survey indicate that there is still work to do regarding visibility and participation in SPCV activities. Less than 25% of both businesses and creators affirmed to have already taken part in activities or events sponsored by SPCV, and almost half of creators were not aware of the society (see Figure 15).

FUTURE PERSPECTIVES FOR THE PORTUGUESE VIDEO GAME INDUSTRY AND ACTIONS TO TAKE

Despite all the difficulties that were highlighted by the survey, the general tone is one of optimism for the future. Asked about their opinion of the future evolution of the economic and financial situation of the Portuguese video game industry a large majority believe it will improve. Also around 60% of businesses and 70% of creators are confident of the future affirmation of Portugal in the global video game industry (see figures 16 and 17).

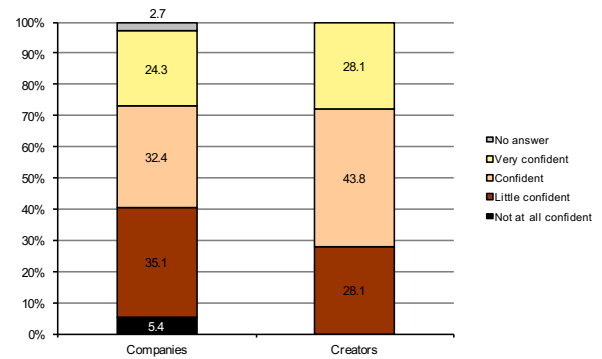


Figure 17. Confidence in the future affirmation of Portugal. Source: Survey to Businesses and Creators (January 2016)

Barrier	Businesses	Creators
Lack of Capital	1st	1st
Lack of public support	2nd	2nd
Human resources	3rd	3rd
Institutional Support	4th	4th
Localization	5th	5th
Facilities	6th	5th

Table 3. Main barriers to the development of the video game industry in Portugal. Source: Survey to Businesses and Creators (January 2016)

We also asked the industry players what were in their opinion the main barriers to the further development and affirmation of the video game industry in Portugal. Both businesses and creators ranked them as described in Table 3. Lack of investment capital was identified as the main factor holding the sector, followed by a lack of public support (regulation, fiscal incentives, etc) , and human resources in third place.

Given the state of the Portuguese video game industry as characterised by the survey and interviews, we have identified a few institutional actions that should be undertaken in order to strengthen the sector.

Know better

The work presented here is pioneer in trying to characterize the industry of video games in Portugal . Given the growing international importance of the sector and its dynamics in Portugal, there is the need to continue and deepen this work in the years ahead. The regularity in the collection of information, as well as increasing the response rate to the survey appear to be the main challenges to the characterization and monitoring of this industry. While the presented results are just descriptive, more data and its evolution along time will allow to extract trends and best practices that will allow better decision making by the industry agents.

Game on, Portugal

The lack of private funding and public support in its different facets (eg. Industry regulation, fiscal and / or financial incentives, etc) are the main barriers to the development of the video game Industry in Portugal. The sector’s potential as a development lever justifies the design of specific public

policies, given the importance of video games as a cultural product with great capacity to export and with high added value as well as its strong contribution to innovation in other sectors. The introduction of public incentives in Portugal (as already happens in other countries where the development of video games is comparable to film-making) could prove to be critical to retain, attract and leverage benefit-generating resources for the national economy. This in order to, first, not to waste the investment made in recent years at the level of higher education in this area and, second, to support the maturation and growth of enterprises and the enormous talent and energy that characterizes them.

The tripple-helix

The triple helix model [3] was developed in the context of knowledge based economies, and it considers three 'sub-dynamic' interactions between Education, Industry and Government. Applying this model, we need, besides public support, to strengthen innovation, differentiation and added value by supporting industry collaboration with the Scientific and Technological System through communication lines that reflect the specific and transdisciplinary aspect of video game innovation. Many successful games start as student projects, exploiting new ideas and technologies from the safe harbour of higher education facilities. We need to find support mechanisms that allow those students to transfer that work and experience to enrich the Portuguese industry.

CONCLUSIONS

This study represents a new approach to understand the dynamism of the Portuguese Video Game industry at a time where a shift towards smaller businesses and indie development has occurred in the global market due to technological evolution. It is an opportunity that a small but growing industry in Portugal is aware, and wants to take advantage off. This industry is constituted mainly by micro enterprises that produce games for the international market. It is a fragile sector, much dependent on own equity for investment funds, and almost totally ignored by the Portuguese state. If Portugal is to take advantage of the moment to create a sustainable industry, several action should be taken. Those have to do with a better knowledge of the Portuguese video game industry, this study being a first contribution to this necessary effort, state industry incentives, namely in what regards funding, and the reinforcement of the connections with the academy.

ACKNOWLEDGMENTS

This work has been partially supported by national funds through Fundação para a Ciência e a Tecnologia (FCT) with reference UID/CEC /50021/2013 and by the EC H2020 project RAGE (Realising an Applied Gaming Eco-System); <http://www.rageproject.eu/>; Grant agreement No 644187.

Disclaimer

This work reflects only the authors' view and that the European Commission is not responsible for any use that may be made of the information it contains.

REFERENCES

1. P. Hollins, Li Yuan, , P. A. Santos, J. Becker, and R. Riestra. 2016a. *D7.1 - Summary Report of Business Models*. Technical Report. RAGE Project.
2. P. Hollins, Li Yuan, , P. A. Santos, J. Becker, and R. Riestra. 2016b. *D7.2 - Summary report of Value Chain analysis*. Technical Report. RAGE Project.
3. H. Leydesdorf, L. Etzkowitz. 1998. The triple helix as a model for innovation studies. *Science and public policy* 25, 3 (1998), 195–203.
4. Microsoft. 2014. *Portugal Game Studios Spotlight*. Microsoft Portugal, Lisboa.
5. Microsoft. 2015. *Portugal Game Studios Spotlight*. Microsoft Portugal, Lisboa.
6. F. Nunes and P. Romeiro. 2014. Game On, Noroeste! Jogar para ganhar na Economia Criativa e Digital.. In *Proceedings of Videojogos 2014, Barcelos, Portugal*. IPCA, SPCV.
7. F. Nunes and P. Romeiro. 2016. Dinâmicas Recentes de Clusterização da Economia Criativa e Digital no Porto Cidade-Região: O Caso da Indústria dos Videojogos. *RPER - Revista Portuguesa de Estudos Regionais* 42 (2016), 21–37.
8. P. A. Santos, P. Romeiro, F. Nunes, and C. Pinheiro. 2016. *Atlas do Setor dos Videojogos em Portugal (#1) 2016*. Technical Report. SPCV and RAGE Project.
9. N. Zagalo. 2013. *Videojogos em Portugal- História, Tecnologia e Arte*. FCA, Lisboa.

GAMING AND VR TECHNOLOGIES, POWERS AND DISCONTENTS

Patrícia Gouveia

Faculdade de Belas Artes da Universidade de Lisboa / Arte Multimédia

Largo da Academia Nacional de Belas-Artes, 1249-058 Lisboa, Portugal

ABSTRACT — This paper uses a communication sciences methodological approach (arts-based research¹) in the way it combines speculative thinking, game design and game theory to the interpretation and future use of new gaming devices and software for Virtual Reality game and play experiences. Starting with the following research questions: Is it possible to combine game engines and virtual reality HMD's to enhance presence? Can these peripheral devices sometimes ruin the gaming experience? The aim of this article is to generate and disseminate knowledge in the game design field, in general, and in the use of new devices applied to game and play environments, in particular.

INDEX TERMS — Virtual reality, oculus rift, gaming experiences and play installations.

I. Introduction

Gaming history is prolific in creating rooms or devices to enhance game experience and to create a perceptual feeling of presence and/or immersion. From Morton Heilig's arcade concept (1962) to Ivan Sutherland's *Head-Mounted Display, The Sword of Damocles* (1965-68) ideas. From Myron Kruger's *Videoplace* (1974) to cave software (cave automatic virtual environment, Electronic Visualization Lab at University of Illinois Chicago, 1996²). The pioneers of virtual reality (VR) wanted to research new possibilities of escaping into alternative worlds, rabbit holes or dream spaces. In the abstract of "Virtopia: Emotional Experiences in Virtual Environments", a 1996 article, from Jacquelyn Ford Morie

and Mike Goslin, is stated "few people create virtual worlds in the context of artistic expression. Key to providing intriguing and engaging worlds is an understanding of the power of emotional content. The authors propose that an emotive response can be elicited in a subject through strategic use of imagery and sound in a virtual environment, which will enhance the sensation of immersion in the simulation and thereby help to compensate for the inadequacies of contemporary technology. Their artwork, *Virtopia*, draws on psychology in its implementation of virtual-reality technologies to produce aurally and visually immersive environments that engage the participant on an emotional level."³ (Goslin & Morie, 1996: online)

Mixed and virtual realities support in the immersive environment our feeling of presence and recreate our sensorial apparatus, a *Holodeck*, a fictional facility featured in the *Star Trek* universe in which our entire body can interact with digital objects or assets. Console and interface pioneers did the rest to make us feel inside a digital world. William A. Higinbotham gave us interaction with a meaningful purpose with *Tennis for Two* (1958). Steve Russell invented the first *shoot'em'up* (*Spacewar!* 1962). Ralph Baer created the *Brown Box* (1966) and Nolan Bushnell the gaming industry (1972). Douglas Engelbart imagined the mouse and the chat room and Alan Kay gave us a fiction to interact with, windows to organize and the programming environment to play with. As David Greelish states in his interview with the author: "Alan Kay is known for the Dynabook — his decades-old vision of a portable suite of hardware, software, programming tools and services which would add up to the ultimate creative environment for kids of all ages. Every modern portable computer reflects elements of the

¹ For more on arts-based research please Cf. O'Donnell, D. (2015), "So You Want to be an Artist: when social scientists don the beret". In <https://medium.com/@darrenodonnell/so-you-want-to-be-an-artist-when-social-scientists-don-the-beret-a9d32f272fb5> (accessed 25.10.2015).

² CAVE was designed and created by Margaret Dolinsky one of the few women working in Virtual Reality. For more information on this please cf. Hrvoje Prpic article named "We Need More Women in Virtual Reality!" In <https://www.linkedin.com/pulse/we-need-more-women-virtual-reality-hrvoje-prpic> (accessed 19.11.2015).

³ Goslin, M. & Morie, J. (1996), "Virtopia": Emotional Experiences in Virtual Environments". In http://www.researchgate.net/publication/275697472_Virtopia_Emotional_Experiences_in_Virtual_Environments [accessed 19.11. 2015].

Dynabook concept — the One Laptop Per Child project's Xo Above All Others — and yet none of them have fully realized the concept which Kay was writing about in the early 1970s.” (Greelish, online: 2013) Dynabook had the intention of merging all the portable devices (tablets, smart phones and personal computers). The casual revolution reinvented vintage devices and created new players (Juul, 2010).

During the nineties virtual reality gaming machines started to appear and the field was divided into two different branches, one, with a focus in art and design installations and, another, connected with entertainment software, mainly digital games. Today we find a rebirth of indie connections between gaming and art installations. (cf. Gouveia, 2013) Andy Bossom and Ben Dunning argued in their book about game industry, “we should consider how the medium is being transformed as video games enter a period of cultural maturity. Away from the gilded halls of the big business that dominate the industry and their AAA titles, indie games have become a thriving playground for fresh ideas, uniquely styled outcomes and laid bare game mechanics (Bossom & Dunning, 2016: 109). We are witnessing a rebirth of short term interactive marketing experiences, art installations, interactive devices applied to health and/or indie games and play installations. Some of these experiments are taking advantage of the strong feeling of presence you can get when using VR Head Mounted Displays like Oculus Rift.

II. Mainstream Culture (consoles) and VR devices and techniques

In the first two decades of the 21st century various types of consoles were made with different goals. Wii (Nintendo, 2006) inserted the human body movement into the game space, using *wiimote* and *nunchuck* as peripheral devices, extensions of their on physical body, players could engage in a mirror like relationship with their avatars. The previous consoles' static bodies became immersed into the gaming environment. It worked well for some kind of game typologies, sports and body related interfaces, but not for all gaming experiences like puzzles, first person shooters or sand boxes, to mention just few examples.

Previously, Playstation I, II and III (Sony, 1994) wanted to detail the image of the body with accuracy and pixel rigor to enhance the feeling of consistency and presence. Motion capture systems and software helped insert the realistic movement into games and animations but sometimes was less consistent with the artistic aims of the product and Computer Generated Imagery (CGI) would be more appropriated for aesthetical purposes (Gouveia *et al.*,

2008; Gouveia, 2010). A “real” faked world imagery, with a strong and interesting plot or story, a cognitively demanding environment, would be enough to create a sense of multiple channels of sensory information (Madigan, 2010; 2015) but the *proprioceptive*, our senses coherent moves according to external stimulus, increased the feeling of consistent action and reaction (Gouveia, 2010).

Nintendo Wii enhanced game characteristics leading to spatial presence. Schematic movements are activated but the sense of agency and the transfer of agency to the virtual space may be a severe amputation which the digital games propagate. The *proprioception*, the sense of knowing one's position in space involves a combination of *somatosensory* perception in which we, as human beings, detect 3D objects by touch (eye-hand coordination, hand position, body movement, visual effects, sound design, conformation, etc.). Since force is necessary for real life but seems to be dispensed in games, the *proprioceptive* and *somatosensory* stimulus stops being symptomatic of this physical goals. There is a dichotomy, for example, in the Wii game box: “acting upon other agents and being acted upon – the active and tense acts of hitting someone virtually benefits to some extend from the ability to actually make punching motions, but the patient-relations involved in boxing match must be left to the audiovisual feedback and since this is comparatively sparse when hitting or blocking in Wii Boxing, the game does not do a very good job of fostering ownership of the virtual body in that situation” (Gregersen, A. & Grodal, T., 2009: 77). And the authors warn: “In other words, players can dance, swordfight, and fish the nights away in the comfort of their living room, but they still get no hugs or kisses” (Ibid., 81).

According to Jamie Madigan, “characteristics of games that facilitate immersion can be grouped into two general categories: those that create a rich mental model of the game environment and those that create consistency between the things in that environment.” (Madigan, online, 2010). Consistent action is a key factor to enhance immersion. Kinect system for Xbox 360 (Microsoft, 2010) continues the same trend of inserting the player body into the game space used in previous gaming technologies. It also contributed to merge experimental game design and play installations (indie game play) and mainstream entertainment products (triple A games). Although Kinect system wasn't to well received among hardcore players it shaped a new trend in indie installations and gaming. The same can happen with VR technologies such as Oculus Rift or similar.

Virtual reality merged with game consoles and engines made gaming more inclusive. Casual games increased the

number of players who can afford to play and gave them more chances to be able to participate in short game sessions and different types of playability (Juul, 2010). Games became available everywhere (tablets, smart phones, personal computers) and to everybody (different ages and genders, ethnicities and tastes, etc.). We can consider that the feeling of being inside the game or the VR space can generate empathy with real people or game characters. We can use VR technologies and gaming devices to affect other people lives and environments. Chris Milk in his TED Talk named “How Virtual Reality can create the ultimate empathy machine” (2015) considers that VR can change and affect the lives of millions of people. (cf. Watercutterin, online: 2015a)

VR can be instrumental to insert participants into experimental documentaries and films where they are integrated into the “other” life. This field is just about to begin. Watercutterin considers, the immediacy of the 360-degree VR view gives the film a poignancy that a news report simply can’t. This technology is being used to immerse interactive participants into human dramas, Ebola disease, in *Waves of Grace*, or *Clouds Over Sidra*, a VR experience about a young Syrian refugee in Jordan. As Watercutterin states, “it’s a powerful message even when read on paper, but when heard while standing amongst the orphans themselves and the graves of some of the more than 4,800 lives Ebola has claimed in Liberia, it’s downright heartbreaking.” (Watercutterin, online: 2015b) The sense of being there with these victims is a powerful tool to create awareness and empathy. Game play devices and VR technologies can be instrumental to help real world problems and to catch new audiences.

III. Story, Texture, Force Feedback, Level Design, Mapping and Architectural Coherence

The haptic experience is connected with our skin surface and in a game we receive force feedback, motion stimulation and vibration. (Bogost, 2008) According to Ian Bogost, in a *Gamasutra* article about the use of texture in digital games, “in the era of 3D computer graphics, texture is a term frequently used in technical talk about video games. Textures are the graphical skins laid atop 3D models so they appear to have surface detail. Texturing techniques like bump mapping and normal mapping use two-dimensional image data to perturb the lighting patterns applied to objects by 3D rendering algorithms to make them appear to have a surface texture that is not actually present in the 3D model itself. These simulate the appearance, but not the behavior or sensation of texture. This is nothing new; The Fine Arts have often done similar. Unlike paintings games are not static scenes or objects -- they are interactive models of experiences. To simulate the behavior, rather than just the appearance of

texture, games have to use more than visual effects.” (Bogost, online, 2008)

Bogost considers that texture in digital games goes far beyond visual effects and that sound design and simulated properties of the physical world (visuals, sounds, coherent movements, etc.) can increase the immersive feeling of texture. Bogost’s statement is consistent with Jamie Madigan’s concept of *multiple channels of sensory information*, which “means simply that the more senses you assault and the more those senses work in tandem, the better. A bird flying overhead is good. Hearing it screech as it does so is better. 3D may also play a role here, and we can all agree that smell-o-vision will herald in a new era of spatial presence” (Madigan, online: 2010; 2015).

Touch is an undeniable factor of gameplay, argues Bogost, and it will be appropriate to force feedback because if “the player still does not *feel* the texture of the road or the brush of the grasses when he plays, but only the cold plastic of the controller” (Bogost, online: 2008) she/he might leave the game experience. The author continues “unlike painting and sculpture (which forbid touch) and music (which cannot accommodate it), video games *require* user participation. (...) Tactile computer interfaces, or haptics, became a consumer industry by the early 1990s, with companies like Immersion developing cheaper, simpler sensors and motors that allowed such devices to be integrated into objects other than the expensive, awkward gloves and vests of dedicated virtual reality labs.” (Ibid., 2008)

In this scenario, following Bogost, Nintendo 64 *Rumble Pak* add-on appeared and that’s why, “we usually call haptic feedback in video games “rumble.” Rumble allows games to create tactile sensations in addition to visual and aural ones. Cars can now seem to bump with the changing texture of asphalt, gravel, dirt. Technically, rumble in contemporary game systems is more or less all alike: motors spin one or more unevenly molded weights in a housing within the body of a controller. But despite the simplicity of rumble, its effects are quite varied: the pulse of a heartbeat signifies health and instils fear in *Silent Hill*; a tackle in *Madden NFL* registers physically as well as visually; the tremor of a gunshot in *Call of Duty* alerts the player to unseen dangers from behind or above; the vibration of the steering wheel in *Gran Turismo* communicates the force of cornering around a hairpin at speed. The subtle signal of a motor signals the cursor entering a button in the *Wii sports* menu screen; a jolt to the hand in *MVP Baseball* alerts the player to an opponent stealing base; a spin of the rumble pack in *The Legend of Zelda: The Ocarina of Time* signifies the loose feel beneath Link’s feet when a treasure is buried beneath the ground he stands upon.” (Ibid., 2008)

For Bogost, *Rumble* increases immersion, makes players feel part of the game and orients them with consistent feedback through the ludic space. For game designer and scholar Ernest Adams “many modern controllers include a vibration feature, which you can use to provide sensory feedback (often called *rumble*) about games events. Although rumble is not technically an audio element, the player can usually hear it as well as feel it.” (Adams, 2010: 231). *Rumble* creates a *synesthetic* effect, when you stimulate one sensory or cognitive pathway, the audio element, it connects with touch and visuals, and you experience sensations in various senses.

A coherent level design can also contribute to the sense of presence in game worlds. For game designers like Jesse Schell, “all a level designer does is arrange the architecture, props, and challenges in a game in ways that are fun and interesting - that is, making sure there is the right level of challenge, the right amount of reward, the right amount of meaningful choice, and all the other things that make a good game.” (Schell, 2008: 343) Game development requires the creation of rich spaces that make us feel immersed in maps that involve locales, stages or missions. Cultural contexts and surroundings influence everything on the screen. Dan Taylor, who has been in the game industry for over 15 years, considered, in 2013, that good level design is based in ten basic principles: i. Fun to navigate; ii. Does not rely on words to tell the story; iii. Tells the player what to do, but not how to do it; iv. Constantly teaches the player something new; v. Surprising; vi. Empowers the player; vii. Allows the player to control the difficulty; viii. Efficient; ix. Creates emotion; x. Driven by your game’s mechanics. According to Taylor, “videogames are driven by interaction, (...) so stage design should be considered as a ‘gameplay delivery system’. That means bringing together artists, stage designers and programmers to work towards the same goal – interdisciplinary communication is vital for success.” (Taylor, online: 2013)

Games can give players new patterns to research and resolve within the game environment, new places to explore and navigate. Jon Brouchoud considers that “the use of hierarchy in architecture is also an important and incredibly useful tool in game design. An architectural experience where all design elements carry the same visual weight isn’t as meaningful or organized as when some elements are differentiated through scale, color, texture, shape, etc.. By contrasting *rhythm* with *hierarchy*, the player can enhance the game’s wayfinding strategy by making it easier for the player to ‘read’ a building to help remember their relative position in space and to understand where important elements of gameplay can be found.” (Brouchoud, online: 2013) And the author

continues, “urban design, and the way building exteriors work together to shape outdoor space, can also play a role in the design of your game. (...) Placing a monument or statue at the end of a visual axis can help orient the player, and make it easier to remember where the player is located in relation to the rest of the level.” (Ibid., online: 2013) A clear visual map and some clear props to orient the player can be very important to create design consistency and to keep the participant immersed within the game environment. Level design can find inspiration in architectural approaches to orient the player in her/his journey (cf. Totten, 2014).

Phillip Robinson, from the Museum of Gaming Research Center in UK, writes, in the abstract of his draft paper titled “Designing games for Virtual Reality: A Valiant Example” that “Virtual Reality games need to be designed with the particular requirements of immersive gaming in mind and they need to play to the strengths of immersion whilst dealing with the unique issues that Virtual Reality games can create. One of the unavoidable problems with current hardware (2015) is simulator sickness and whilst new hardware will bring inherent improvements good game design now is actually more important and this foundation will be built upon when consumer devices are available (...)” (Robinson, online: 2015)

Story, texture, force feedback, level design, mapping and architectural coherence can contribute to the feeling of immersion in game spaces. Is it possible that the use of Oculus Rift or similar HMDs can jeopardize this sensation by imposing too much proximity in our sensorial apparatus? Besides Oculus Rift, owned by Facebook, other companies, like Sony, Valve, Samsung, Microsoft or Google, are creating similar devices these days. VR technologies will become part of our future gaming experience. Virtual reality is everywhere again, states Brenda Laurel, clarifying what VR is, i. e., i. a complete surround environment; ii. affordances for depth perception and motion parallax; iii. spatialized audio, not just stereo; iv. affordances for tracking the participant’s direction of motion distinct from the direction of gaze; v. the participant’s sensorium as the camera; vi. natural gesture and movement; vii. affordances for narrative construction; viii. the principle of action.” (Laurel, 2016: online)

Although “great technical improvements”, argues Laurel, “are making great progress; e. g., frame rate, visual convergence, 3D modeling, and fine-tuned body tracking through video or other means. My intent is to describe in specific terms the formal and structural aspects of a particular form that was and is called Virtual Reality. I also want to warn younger folks of the consequences of stretching a name too thin. Back in the 1990s, that’s

exactly what happened - and the form, along with discoveries of those who created it - largely disappeared. Let's be mindful of that this time around." (Ibid., 2016: online)

IV. Oculus Rift VR Experiences and the impact of virtual reality

Two Oculus Rift prototypes were made available for purchase as "development kits" (DK1, in 2012; DK2 in 2014). This was an opportunity for game developers and producers to explore and design game products oriented for Oculus Rift in its release date. At Noroff University College, in Norway, we had the chance of working with this technology and to engage students in some experimental projects using one of these prototypes. Our aim was to use an integrated arts-based research methodology, following the previously quoted article of Darren O'Donnell, where we merge knowledge production and dissemination with a participatory strategy: "knowledge analysis that happens naturally in an embodied way through participation. This integration of production and dissemination, plus the addition of the analysis phase — typically missing from art-based methods (usually occurring with the application of non-artistic methods) — puts socially-based art works in a decisively strong position." (O'Donnell, online, 2015)

Raymond Hansen, a bachelor student in Interactive Media at Noroff University College, presented a final thesis research focused on the possibility of creating Appealing Architectural Visualizations using 3D animations made on Maya Software, Oculus Rift development kit and the game engine Unreal Engine 4. This student found that the biggest obstacle against VR experience is motion sickness. He reported, on his final thesis, that several people were complaining about getting sick within minutes of using the Oculus Rift, but some people don't react at all. The student found himself sick, nauseated for almost six hours, after testing Oculus Rift during 30 minutes (Hansen, 2015: 29). This research is consistent with our own tests with colleagues and students at the Faculty of Interactive Media (Games and Animation) at Noroff University College.

Raymond Hansen also found that both the resolution and the ergonomics of Oculus Rift have to be increased. According to this student, "The currently available Development Kit (DK2) feels kind of heavy, and after a few minutes of usage, sweat does build up, because of the intensity of the experience, and that is not that appealing." (Hansen, 2015: 30).

enables the sensation of presence – the feeling as though you're actually there. The magic of presence changes everything. You've never experienced immersion like this." (Oculus VR website, 2015) For Oculus VR, "seeing is believing" and "The Rift uses state of the art displays and optics designed specifically for VR. Its high refresh rate and low-persistence display work together with its custom optics system to provide incredible visual fidelity and an immersive, wide field of view." With an "Advanced and refined design, from the moment you pick up the Rift, you'll feel and see the attention to detail that went into its design and construction. Customizable, comfortable, adaptable, and beautiful, the Rift is technology and design as remarkable as the experiences it enables" (Ibid., 2015). And the company is also "introducing Oculus Touch, a new pair of tracked controllers that let you take your VR games and experiences further than ever before" (Ibid., 2015).

According to Hansen, "experiencing Virtual Reality with the Oculus Rift is really immersive. The brain is actually fooled to believe what we see is real, but that is also part of the problem. If the brain is "sensing" that we are moving in a virtual world, while we're sitting still, it's easy to become motion sick – which is the biggest problem with virtual reality devices today. The motion sickness [sensation is not a pleasant one]. The resolution is also too low, as each OLED display only has a resolution of 960 x 1080 per eye. It's just not enough for a smooth experience, but Oculus has already done something with this with the Crescent Bay version [more recent versions]. People who have tried it reported that the resolution has been increased, but Oculus themselves has not released any official information on what the resolution in Crescent Bay is, and the resolution will probably increase again in the upcoming consumer version." (Hansen, 2015: 26).

The impact of virtual reality devices on human behavior is still a concern for researchers. According to the article "Long-term effects of virtual reality use need more research, say scientists", from Nicola Davis, there is plenty of work to be done. Virtual reality systems can be used in a host of therapeutic situations, from helping those living with post traumatic stress disorder to those suffering from depression, but we need to consider that maybe this technology could have detrimental effects and it is better to stop using the headset if you feel unwell. Good advises could be don't use this technologies for more than 30 minutes and make sure someone is keeping an eye on you.

The Oculus Rift website promises "the magic of presence" where "The Rift's advanced display technology combined with its precise, low-latency constellation tracking system

V. Vertigo play (*Ilinx*)

Oculus Rift VR experiences look like thematic parks or installations which deal with our perceptual apparatus. French game scholar, Roger Callois, in his attempt to classify game typologies called these strong sensations or vertigo experiences *Ilinx*. *Ilinx* is the Greek word for spin, twist, from which derives the Greek word for dizziness (vertigo / ilingos). *Ilinx* is disorder that takes an organic or psychological form. A sense of brutality and intensity capable of troubling human beings. The industrial revolution was lavish in machine construction and powerful devices and vertigo becomes a kind of game / play (Callois, 1961: 24-25). Sometimes these devices make us feel dizzy, let us jump into imaginary worlds, other times, the cinematic atmosphere can make us travel to real worlds, like in the above mentioned VR experiences *Waves of Grace* about Ebola, or *Clouds Over Sidra*, about a young Syrian refugee in Jordan. The experience could be so intense that it is not bearable for too long. As Callois argues in his previously quoted book, *Man, Games and Play*, *Ilinx* is more play (*paidia*) than game (*ludus*).

At this stage we can now answer our research questions which were the following. Is it possible to combine game engines and virtual reality HMDs to enhance presence/immersion? Can these peripheral devices ruin the gaming experience? We consider that the use of VR devices such as the Oculus Rift can be appropriated in short term play experiences but that they can be a source of discomfort for long term gaming experiences where the player should focus in other components of game play. The intensity of VR experiences could eventually make the player lose focus of the story and the consistency of the game world can be ruined. The capacity to directly interact with the game world can deceive players with a sense of real agency but if the experience is not short it can leave participants with a sense of nausea or discomfort. Short sessions are also available in thematic parks to make sure people can recover from such a strong vertigo. Games belong to a different branch of meaningful experiences and even if they can be combined with play moments, for Oculus Rift or other VR technologies use, they will still need some breathing room to let players solve puzzles, accomplish missions, explore maps and stories. With or without the strong effects of VR, vertigo and *paidia*, for now games will keep their path through *ludus* experiences, with chance and competition, joining only simulation and *ilinx* if appropriated by their game design.

According to Phillip Robinson, in his previously quoted draft paper, “VR games need to be designed with the particular requirements of immersive gaming in mind and

they need to play to the strengths of immersion whilst dealing with the unique issues that VR games create. One of the unavoidable problems with current hardware (2015) is simulator sickness and no matter how good a game is, if it makes players feel sick they are unlikely to embrace it. (Robinson, online: 2015)

Robinson considers that “the problem is that as the level of immersion that a VR game offer increases it has a negative effect on the player’s brain. The immersive nature of VR allows players to become convinced that they are in motion and they begin to experience an increasing amount of sensory conflict.” (Ibid.: 2015) This is consistent with Raymond Hansen’s experiments and exemplifies how important it is to focus on game design and to research some technological effects to see if they are supporting our creative ideas instead of dominating them. The game design should be consistent and the game concept and art should be coherent with the game technological apparatus and the player feeling of agency. For Robinson, “improved hardware is on the way, the Vive headset and the commercial release of the Oculus Rift both have faster refresh rates and higher resolutions than current development hardware and this will lessen the effects of simulator sickness. Whilst new hardware will bring inherent improvements good game design is actually more important and here the game *Valiant* by “Offpeak Games” [3] has produced an experience worthy of examination.” (Ibid.: 2015)

Following Phillip Robinson’s and Raymond Hansen’s research in using Oculus Rift for different purposes, i.e., to explore in a fun way architectural places or to play games, we can suggest that we need to take into account our body congruence with the game engine and external devices in harmony with game design decisions (characters, worlds, actions, reactions, textures, visual details, maps, sounds, among other factors) and player agency. As Robinson states, “simulator sickness is a natural response by our physiology and in the world of VR it is immersion that we are after. Immersion is key to the experience and convincing our brain that what it is seeing and experiencing is reality is the whole purpose; it is a trick and our bodies do not always respond well to being tricked. (Ibid.: 2015) To play with a purpose is a key factor to the feeling of agency and/or presence.

VI. Play with a Purpose (Indie Games and Marketing Strategies using VR technologies)

Christina Campodonico’s article “Play with a Purpose” at the Argonaut online published last October (2015) shows a IndieCade 2014 show reel where we can find various interactive experiences with VR devices such as Oculus Rift. Indie games promote multiple disruptive and old

school techniques as we stated elsewhere (Gouveia, 2013) and it's a territory open to experimentation. According to Campodonico, "IndieCade challenges such stereotypes about the world of gaming by celebrating independent games and game designers from all over the country and the world. From board games to scenarios set in virtual reality, IndieCade welcomes games of all stripes – as long as they might open your mind." (Campodonico, online: 2015) Innovated experiences such as those promoted by IndieCade can accommodate Virtual Reality devices in a creative perspective.

Marketing strategists can also integrate VR in their product promotion. Tracy Brady in "8 things you need to know before launching a virtual reality brand experience" states that "Virtual reality is fast becoming one of advertising's buzziest tactics, and technologies like Oculus Rift have made it possible to create rich, multisensory virtual experiences. Last year during the World Cup, Coca-Cola held a virtual reality experience [created by Ydreams Brazil] for fans, giving them a peek into locker rooms in Brazil's Maracanã Stadium. HBO has also been an early adopter, replicating a massive wall of ice from its hit television series "Game of Thrones." The wall of ice allows users to experience wind and floor shaking as they travel in a virtual elevator along the icy face." (Brady, online: 2015)

Tracy Brady considers that we should take into account eight things when we start design a VR experience. First, we should think about the all strategy and to adapt technology to the goals of the brand. Second, we have to watch out for motion sickness. Third, technology is key, "the tech experience is dependent upon having a processor powerful enough to support the experience from a graphical standpoint. For something like Trailscape [Merrell's "Trailscape," the immersive sensory experience debuted at Sundance Festival in 2015], the depth of the 3D environment and the speed at which the engine needed to react to human head/eye movements couldn't be supported by a consumer-grade computer and requires a custom-build CPU in order to run." (Ibid., 2015) Fourth, "expect the unexpected", debug the system. Fifth, "Go beyond the visual: For a truly immersive sensory experience, going beyond the visual means considering factors like sound design, tactile scenic components, and "4D" elements, such as vibration and artificial wind. These are huge factors in taking full advantage of users' senses, particularly for the exploration factor of Trailscape. Using a VR headset as part of an experience isn't news by itself anymore; it's now about context. In many ways, the hardware/tech story is more about the content and using VR headsets in ways that make sense for the medium and not just using the technology for its own sake." (Ibid., 2015) Sixth, "Optimize your location: The financial

investment in something like this can be significant, and the number of people who can experience it is often low, so it's important to have a plan to get the installation in front of as many eyes as possible. (Ibid., 2015) Seventh, "Keep costs in mind: As costs add up, the number of active installations you can afford may have to be minimized (...)." Finally, "go for curb appeal: It's crucial to think about the "curb appeal" of the experience. If someone walks in off the street, what do they see? Does it look fun or impressive? Since a relatively low number of people will be able to participate, and explaining VR doesn't do it justice, it's important to keep in mind that sharing photos of the installation and your physical setup around it will be how most people will spread the word about what you're doing." (Ibid., 2015) For the above quoted author, "even though we're still in the early days of VR for marketing, it's easy and exciting to imagine its potential." (Ibid., 2015)

VII. CONCLUSION

According to our research Oculus Rift can increase the feeling of presence (immersion) in indie games and installations but it might jeopardize other long term gaming experiences. The design of these play environments need to take into account several factors like coherent texture, level design, consistent human machine interaction and map planning. Virtual Reality remains a vertigo (*ilinx*) territory, where users are so immersed in the digital environment that they can play with their own sensorial apparatus, but the all experience is so intense that it can ruin other game design attributes such as plot or story consistency, meaningful interaction and game play, game mechanics, among others. Besides technical effects for mainstream games Oculus Rift can enrich play experiences in indie installations, documentaries about reality and marketing strategies. Future research will probably solve the Oculus Rift Development Kit problems, like high-resolution imagery and reducing latency, but a lot more tests need to be made to better connect the Oculus with game engines and web browsers.

In Anna Burzlaff's text "Aaron Koblin on VR Storytelling" (2015) the artist states that what got him excited about VR applications and tools was the potential to change the way we communicate and think. For Koblin today we are only scratching the surface of this territory and we can achieve an unprecedented degree of intimacy even if "someone unfamiliar with VR might find unusual given the clunky headsets, wires, and other technical wizardry (...)." (Burzlaff, online, 2015) The visceral side of VR involves you in a powerful relation with your senses where, says Koblin quoted in Burzlaff, "you can't look away, you're fully immersed and vulnerable and that's a powerful thing. (...) Powerful VR content "has the

potential to connect its users to others and engage them in emotional experiences that may otherwise be too far removed from their immediate environment. (...) The difference is that being a part of virtual reality is not an ephemeral experience between your imagination and the storyteller. It's actually a much more visceral presence and experience when you're a part of the environment. That means there are also greater opportunities and a plethora of challenges as well." (Ibid., 2015)

For Koblin quoted in Burzlaff, "The applications of VR are definitely not restricted to entertainment. Some of the most exciting potentials lie in medicine, where doctors can trail surgery in virtual environments, and psychology, where immersive therapy has already been successful in treating post-traumatic stress disorder in soldiers. It could also prove invaluable for brands." (Ibid., 2015) Indie games will also take advantage of this new field and new tools and techniques will be created to engage players and their senses all over the globe. Oculus Rift (Oculus VR) can increase the feeling of presence (immersion) but it might jeopardize engagement in some gaming experiences. Even though the powers of VR may not lie in hardcore gaming they might open bright new possibilities for exploration elsewhere, may it be medicine, art worlds, documentaries or indie playful installations. VR can be a truly meaningful experience.

REFERENCES

- [1] Adams, E., (2010), *Fundamentals of Game Design*, Second Edition, New Riders, Berkeley.
- [2] Bogost, I. (2008), "Persuasive Games: Texture". In http://www.gamasutra.com/view/feature/132053/persuasiv_e_games_texture.php. (accessed 20.10.2015).
- [3] Bossom, A., & Dunning, B., (2016), *Video Games, An introduction to the Industry*. Creative Careers, Bloomsbury. London, Oxford, NY (...).
- [4] Brady, T. (2015), "8 things you need to know before launching a virtual reality brand experience". In <http://venturebeat.com/2015/03/08/8-things-you-need-to-know-before-launching-a-virtual-reality-brand-experience/> (accessed 20.10.2015).
- [5] Brouchoud, J. (2013), "The Importance of Architecture in Video Games and Virtual Worlds". In <http://archvirtual.com/2013/02/09/the-importance-of-architecture-in-video-games-and-virtual-worlds/#.VDu6hWB01Mt> (accessed 20.10.2015).
- [6] Burzlaff, A. (2015), text "Aaron Koblin on VR Storytelling". In <https://www.prote.in/en/profiles/aaron-koblin-on-vr-storytelling> (accessed 25.10.2015).
- [7] Caillois, R. ([1958] 2001), *Man, Play and Games*, University of Illinois Press, Urbana and Chicago.
- [8] Davis, N., (2016), "Long-term effects of virtual reality use need more research, say scientists". In <https://www.theguardian.com/technology/2016/mar/19/long-term-effects-of-virtual-reality-use-need-more-research-say-scientists>
- [9] Campodonico, C. (2015). "Play with a Purpose". In <http://argonautnews.com/play-with-a-purpose/> (accessed 20.10.2015).
- [10] Gouveia *et al.* (2008), "Realism in Gameplay: Digital Fiction and Embodiment", *ACM 2008 Proceeding of the 2nd ACM international workshop on Story representation, mechanism and context*, Vancouver, British Columbia, Canada, ISBN:978-1-60558-315-0, pp. 1-8.
- [11] Gouveia, P. (2010), *Digital Arts and Games, Aesthetics and Design of Playful Experiences [Artes e Jogos Digitais, Estética e Design da Experiência Lúdica]*, Ed. Universitárias Lusófonas (11). ISBN: 978-972-8881-84-9, Lisbon.
- [12] Gouveia, P. (2013), "Transmedia experience and the independent game industry: the renaissance of old school visual and artistic techniques" [Experiência transmedia e a indústria de jogos independente (indie): o "renascimento" de expressões artísticas e visuais do tipo "old school"], Petry, A. S., Trindade, E., Petry, L. C., Llano, N., (2013), (ed. by), *Communication and Social Antropology Seminar [Comunicação e Antropologia Social]*, Internacional Seminar from the ICCI Network [Seminário Internacional de Rede ICCI Imagens da Cultura Cultura das Imagens]. ECA, USP, São Paulo, Brazil, pp. 78-89. (e-book). ISBN 97885879632.In http://www2.eca.usp.br/redeicci/ebook/ebook_ICCI_2013.pdf [accessed 19.11. 2015].
- [13] Goslin, M. & Morie, J. (1996), "Virtopia": Emotional Experiences in Virtual Environments". In http://www.researchgate.net/publication/275697472_Virtopia_Emotional_Experiences_in_Virtual_Environments [accessed 19.11. 2015].
- [14] Greelish, D. (2013), "An Interview with Computing Pioneer Alan Kay". Time Interview. In <http://techland.time.com/2013/04/02/an-interview-with-computing-pioneer-alan-kay> (accessed 20.10.2015).
- [15] Gregersen, A. & Grodal, T. (2009) "Embodiment and Interface", *The Video Game Theory Reader 2*, ed. M. J. P. Wolf & B. Perron, Routledge, NY and London, pp. 65-83.
- [16] Hansen, R. (2015), *Appealing Architectural Visualization in Virtual Reality with Game Engines*, Faculty of Interactive Media (Games and Animation), Noroff University College (unpublished thesis).

- [17] Juul, J., (2010), *A Casual Revolution, Reinventing Video Games and Their Players*, Cambridge, Mass.: MIT Press.
- [18] Laurel, B., (2016), "What is Virtual Reality?". In <https://medium.com/@blaurel/what-is-virtual-reality-77b876d829ba#.p5ytp4867> (accessed 18.09.2016).
- [19] Madigan, J. (2015), *Getting Gamers: The Psychology of Video Games and Their Impact On Those Who Play Them*. Rowman & Littlefield Publishers.
- [20] Madigan, J. (2010), "The Psychology of Immersion in Video Games". In <http://www.psychologyofgames.com/2010/07/the-psychology-of-immersion-in-video-games/> (accessed 20.10.2015).
- [21] Milk, C. (2015), "How Virtual Reality can create the ultimate empathy machine". In "How Virtual Reality can create the ultimate empathy machine" (accessed 22.10.2015).
- [22] O'Donnell, D. (2015), "So You Want to be an Artist: when social scientists don the beret". In <https://medium.com/@darrenodonnell/so-you-want-to-be-an-artist-when-social-scientists-don-the-beret-a9d32f272fb5> (accessed 25.10.2015).
- [23] Prpic, H. (2015), "We Need More Women in Virtual Reality!" In <https://www.linkedin.com/pulse/we-need-more-women-virtual-reality-hrvoje-prpic> (accessed 19.11.2015).
- [24] Robinson, P., (2015), "Designing games for Virtual Reality: A Valiant Example." In http://www.museumofgaming.org.uk/papers/games_design_for_virtual_reality.pdf (accessed 05.01.2016).
- [25] Schell, J., (2008), *The Art of Game Design, a book of lenses*, Morgan Kaufman, Elsevier, Amsterdam.
- [26] Taylor, D. (2013), "Ten Principles of Good Level Design (Part 1 and 2)". In http://www.gamasutra.com/blogs/DanTaylor/20130929/196791/Ten_Principles_of_Good_Level_Design_Part_1.php and http://www.gamasutra.com/blogs/DanTaylor/20131006/197209/Ten_Principles_of_Good_Level_Design_Part_2.php. (accessed 20.10.2015).
- [27] Totten, C. W., (2014), *An Architectural Approach to Level Design*, CRC Pres, A K. Peters Books, London, NY.
- [28] VV.AA. (2015), "Rift, Next-generation Virtual Reality". In <https://www.oculus.com/en-us/rift/> (accessed 20.10.2015).
- [29] Watercutterin, A. (2015a), "Apple Showcases Its VR Dreams – With a U2 Video" (Wired). In <http://www.wired.com/tag/chris-milk/> (accessed 22.10.2015).
- [30] Watercutterin, A. (2015b), "This Ebola Documentary Shows VR Film's Radical Potential". In <http://www.wired.com/2015/09/vr-ebola-film/> (accessed 22.10.2015).

MiniPool: Real-time artificial player for an 8-Ball video game

David Silva

INESC-ID and Instituto Superior Técnico,
Universidade de Lisboa
david.d.silva@tecnico.ulisboa.pt

Rui Prada

INESC-ID and Instituto Superior Técnico,
Universidade de Lisboa
rui.prada@tecnico.ulisboa.pt

ABSTRACT

The importance of artificial intelligence in games has been growing over the years due to the need of giving the player a more challenging experience. Games like 8-Ball offer many interesting challenges to both communities of AI and optimization due to the continuous and stochastic characteristics of the domain. To succeed, a player must be able to plan the best sequence of shots and execute a shot with accuracy and precision, so he does not lose the turn. There are already several artificial players, however they tend to take more than 30 seconds to select and execute a shot. Under a videogame setting, a player would give up playing the game if he had to wait that long for his turn. To address this problem, we propose a real-time solution for an 8-Ball artificial player using a Monte-Carlo Expectimax hybrid search algorithm with ray tracing techniques.

Author Keywords

Artificial Player; Billiards; 8-Ball; Stochastic Game; Video Game; Real-time

INTRODUCTION

Artificial players for 8-Ball games have been a topic of investigation due to interesting aspects that cannot be solved using the traditional methods of the classic games [5]. Having a continuous and stochastic domain makes it possible to have an infinite number of states and actions. It is also difficult to predict the resulting state of an action due to perturbations on the environment that cannot be controlled by the player (it is very unlikely that two shots with the same parameters lead to the same resulting states).

In the last years a competition called *Pool Computer Olympiads* has been taking place. In this competition participants had to develop an artificial billiards player and compete

with each other. All these artificial players, which will be mentioned later in Section 3, focus on different aspects of the game and explore different points of view to overcome the design challenges found in 8-Ball.

One possible application of the state of the art in computer billiards is video games. The games industry has been growing over the years. According to the Entertainment Software Association¹ statistics in 2014, 155 million Americans play video games. Developing artificial intelligence for games brings other interesting challenges such as dealing with limited resources, providing a response in real time and balancing the skills. A professional pool player needs to have a good planning and understanding of the table state to make the best decision. For the artificial player to be able to do this and compete with the best human pool players, it will need the tools to plan the best sequence of shots, support a large variety of shot patterns, mechanisms to evaluate and find good reposition zones for the cue ball and methods to optimize the shot parameters under stochastic environments. The artificial players already developed for 8-Ball tend to take more than 30 seconds to plan the best shot to be executed. This delay on the response would probably make most players give up playing the game. In the context of this kind of games, players are expecting a real time response from the opponent.

The main focus of this work is to develop a real-time artificial 8-Ball player capable of competing with the best players while having a good balance between skill and resources used.

The rest of this document is structured as follows: first, in Section 2, we present a short background of 8-Ball and an overview of the characteristics of the game where the proposed solution will be tested. In Section 3, we introduce and discuss some of the related work already done. In Section 4, we give an overview of the proposed solution as well as some explanation of the key elements of MiniPool. The Section 5 contains all the experimental results and analysis of the individual contributions of each component of Minipool. Finally, in Section 6, there is an overall discussion and future work.

Videojogos'16, Nov 24–25, 2016, Covilhã, Portugal

Frutuoso Silva e Pedro Santos (eds.) ISBN 978-989-20-7148-0.

¹<http://www.theesa.com/>

BACKGROUND

To provide a general idea of the challenges present in the 8-Ball game, we will describe the general rules used in the game, as well as the shot parameters and the most common shot patterns used by professional players.

8-Ball Rules

8-Ball belongs to the Pool-Billiards games family. It is a turn-based game played by two players on a rectangular pool table with 6 pockets and 16 balls (7 solid, 7 striped, the 8 ball and the cue ball). The game begins with a player striking the cue ball anywhere behind the *headstring* (line with a semi-circle, illustrated in Figure 1) towards the cluster of the rest of the balls (called *break shot*).

A player can only strike the cue ball and only using the cue stick. To pocket a ball, the player can try to hit the object ball directly with the cue ball, hoping that the velocity gain from the collision is enough to make the object ball enter a pocket, or using collisions with other balls or rails to reach that object ball. The first ball to enter a pocket determines which ball type (solid or striped) the player needs to seek, having the opponent to seek the remaining type. The cue ball needs to hit a ball when stroked and that ball has to be of the type that the current player is seeking, otherwise the player is committing a *foul*. A *foul* also occurs when the cue ball enters a pocket.

When a ball enters a pocket legally, the player keeps the turn and must shoot again from the cue ball current position, otherwise the opponent gets the turn. In the case of a *foul*, the opponent also gets a *ball-in-hand*, which gives him the opportunity to place the cue ball anywhere on the table.

When the last ball of the current player enters the pocket, he needs to seek the 8 ball. When the 8 ball enters a pocket this way, and only at this situation, the player wins the game, but if it enters a pocket in any other situation the player loses the game immediately.

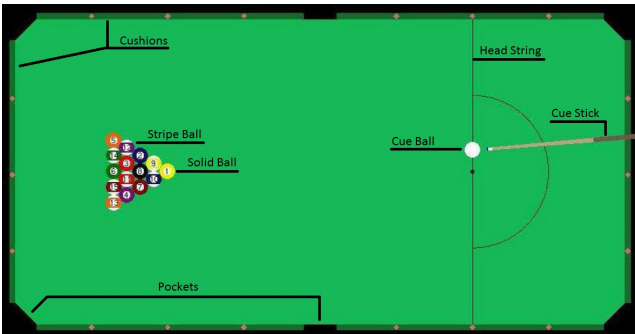


Figure 1: Initial pool table layout

Shot Parameters

In every Billiards variant, a shot is defined by five continuous parameters (illustrated in Figure 2):

- ϕ : Aiming angle,
- θ : Cue stick elevation angle,
- V : Initial cue stick impact velocity,

- a and b : Coordinates of the cue stick impact point on the cue ball.

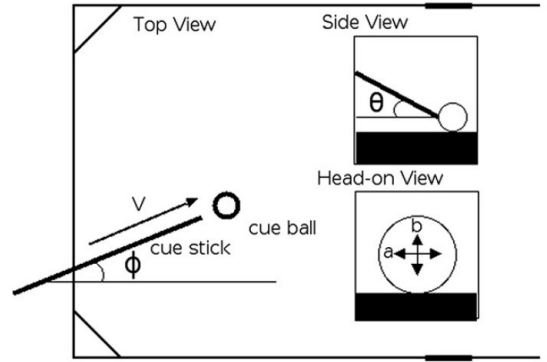


Figure 2: Parameters that define a shot in Billiards [17]

Shot Types

Pocketing, striking and kissing are some of the events that can occur in a regular 8-Ball match. The event that happens when a ball enters a pocket is called *pocketing*; a collision between a moving ball and a stationary ball with the purpose of moving the stationary ball is called *strike*; a shot whose purpose is just to adjust the trajectory of the moving ball is called a *kiss*. A shot can also consist in a combination of events. By combining them, human players can distinguish a variety of shot types. The following are the most common [5] (illustrated in Figure 3):

- **Break shot:** Initial shot, which has the purpose of dispersing the initial cluster of balls.
- **Direct shot:** The object ball is directly hit by the cue ball towards a pocket, without any other collisions involved.
- **Bank shot:** The rail is used to maneuver the object ball towards a pocket.
- **Kick shot:** The rail is used to maneuver the cue ball towards the object ball.
- **Combination shot:** A collision with another ball is used to attempt to pocket the object ball.
- **Pulk shot:** Similar to combination shot but the two object balls are very close to each other and align in a way that they are pointing to a pocket.
- **Kiss shot:** An additional ball is used to adjust the trajectory of another ball.
- **Safe shot:** This type of shot is used when the player assumes that he is more likely to lose the turn. Its purpose is to reposition the cue ball in such way that it makes it difficult for the opponent to continue.

RELATED WORK

Currently, there are already several artificial 8-Ball players; PickPocket [17, 18], CueCard [2, 1, 4] and PoolMaster [10, 11, 13, 14, 15], which participated in the *Pool Computer Olympiads*, and JPool [5].

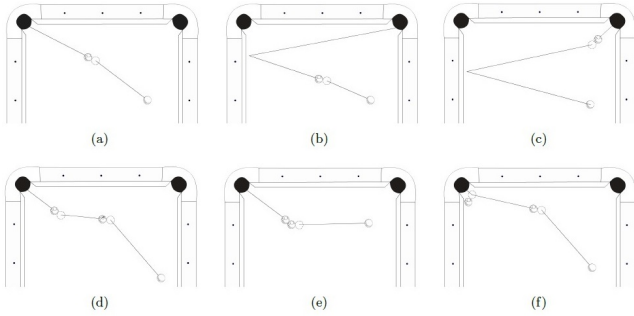


Figure 3: Shot types: (a) direct shot, (b) bank shot, (c) kick shot, (d) combination shot, (e) pulk shot, (f) kiss shot [5]

To better understand the differences among them, in this section, we will go through the most important topics, starting with a possible game model for 8-Ball found by Christopher Archibald *et al.*

Billiards Game Model

Christopher Archibald [3, 4] proved that billiards has a *pure stationary Markov perfect equilibrium* [7]. This means that when a player is selecting a shot to execute, he only needs to think about the current state of the game to get the optimal shot. At some state s of the game, he will not change his strategy concerning on how that state was reached, whether the game is at the beginning or at the end or whether the opponent is good or bad. When selecting a shot, we want it to pocket a ball and have the cue ball at a good reposition to continue playing. If that is not possible under our strategy, we want the cue ball to be at a position where a successful shot would be as difficult as possible for any opponent. With this in mind, we have theoretical proof that we only need to explore shots based on the current and future states (for position play) of the game to reach the optimal shot. Trying to understand who is our opponent and what are his actions on the game will not give us better results. If a strategy is optimal it will remain optimal regardless the opponent actions and strategy. However, the proposed model remains intractable due to the action space being continuous. To compute the value of a state, we would need to try all the possible actions, which are infinite. Thus, any approach still needs to perform an intelligent search space partitioning to overcome this problem.

Search Algorithms

For the particular case of 8-ball and considering stochastic environments, PickPocket [17, 18] and JPool [5] suggested *Expectimax* and *Monte-Carlo* as possible search algorithms.

Expectimax generates chance nodes for every action with a stochastic outcome. These chance nodes will be evaluated with their probability of occurrence. In the case of 8-Ball, a direct approach would mean a sum of over an infinite number of outcomes, each with a minuscule probability of occurring.

Monte-Carlo was used in almost all the artificial players that will be explained in Section 3.3, with the only exception of PoolMaster, since *Monte-Carlo* by sampling shots avoids the

limitation of *Expectimax* and gives the developer more control over the algorithm complexity.

PoolMaster uses a heuristic based search algorithm to select the best sequence of shots. It clusters balls with the *K-Means* algorithm [16] to improve the search and explore less riskier shots first.

Shot Generation

8-ball has a continuous and stochastic domain nature, so it is impossible to enumerate all the possible shots. Generating only the most relevant shots for a particular situation is the key for an intelligent search space partitioning that allows to improve the overall performance of the program.

Since the shot generator algorithm differs for each one of the artificial players, we will present them individually.

PickPocket

Generates shots one type at a time in an increasing order of difficulty. Variations are generated by perturbing the original shot with the base velocity retrieved from a precomputed table. The break shot parameters are selected by sampling 200 shot variations and selecting the one which returns the best results. Safe shots are generated by perturbing V and ϕ and evaluating in the opponent's perspective. For Ball-in-hand situation, the table is discretized in a grid and every cell is assigned with the value of the best shot as if the ball was there, and then a *Hill-Climbing search* is performed in several random cells to find a local maximum.

CueCard

Similar to PickPocket, but it does not prioritize the shot types. It clusters similar resulting states with *K-Means* to reduce the state space. The Break Shot used was precomputed. The Ball-in-hand is similar to PickPocket but before discretizing the table, it tries to place the cue ball where the ghost-ball would be (see Section 3.4).

PoolMaster

The focus of PoolMaster lies on position play. First it generates all pairs ball-pocket possible given the current state of the table, then it analyzes the table for the possible next shots. Once this information is gathered, it calls an optimization algorithm to minimize an objective function that takes into account the distance to the next shot, as well as pocketing the target ball.

JPool

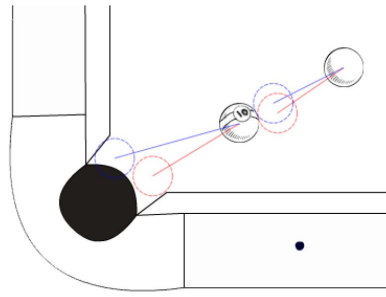
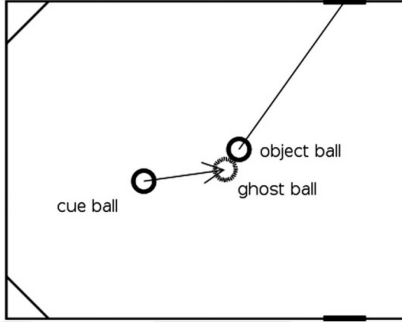
Takes a different approach, given that it models a shot as a series of steps like a tree, not limiting itself to predefined shot types. The break shot parameters were precomputed. For position play, JPool creates polygons around every ball where it would be a good place to pocket and then does a line-polygon overlap detection using the trajectory of the cue ball at maximum speed. The crossing zones are rated as if the cue ball was there and the cue ball is aimed to reach these areas. The rest of the parameters are discretized.

Aiming

PickPocket [17, 18] and CueCard [2, 4] use the traditional concept called *ghost ball* (illustrated in Figure 4a). If the cue ball is aimed in such a way that it hits the object ball in the

position of the ghost ball, the object ball will travel in the direction of the target position.

PoolMaster [10, 11, 13, 14, 15] and JPool [5] use the same concept in a different way. Instead of aiming the object ball to the center of the pocket, they aim it to its limits, which gives them two ghost ball positions (illustrated in Figure 4). These leftmost and rightmost are adjusted to take into account the possible obstacles in the way, so, if the ball does not fit between these margins, the shot will be impossible [6].



(b) Leftmost and rightmost concept [5]
Figure 4: Aiming concepts

Evaluation Function

A search algorithm evaluates shots to differentiate them and selects the best one for execution. There are several ways of measuring and differentiating shots from each other.

JPool [5] uses an algorithm based on Monte-Carlo search with a sample size of 400. The leaves are evaluated using a sum of several heuristics advised by a professional player [12] such as the quality of the current cue ball position, the number of balls in game and the difficulty of pocketing the other balls.

PickPocket [17, 18] and CueCard [2, 4] also use a Monte-Carlo search base algorithm, with 15 and 25 to 100 samples (depending on the time available), respectively. The leaves are evaluated using the sum of the probability of success of the best 3 shots retrieved from the precomputed table.

PoolMaster [10, 11, 13, 14, 15] calculates the value of a node using a function that takes into account the quality of the cue ball position, the probability of being in that position zone and the range of successful parameters with a sample of 15.

Shing Chua *et al.* explained in [8, 9] the calculation of the shot difficulty using fuzzy logic. The fuzzy sets were defined for

the distance traveled by the cue ball before the collision with the target ball, the distance from the target ball to the pocket and the cut angle. They infer the rule for the specific shot situation during runtime. Only direct, bank and combination shots are considered and they are prioritized in this order.

IMPLEMENTATION

In this section, we will explain how we explored and implemented the main architectural components of our artificial player: search algorithm, shot generator and evaluation function.

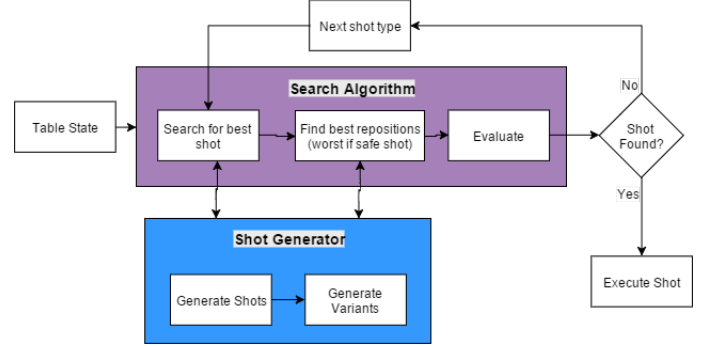


Figure 5: General architecture of MiniPool

Search Algorithm

The search algorithm is responsible for selecting the best sequence of shots. In this layer of the artificial player program, it is crucial to only explore the most relevant nodes of the tree to reach a solution, specially when the time constraints are tight.

Monte Carlo was selected as base for the search algorithm since it works very well in the 8-Ball domain, as it can be seen in the artificial players studied in Section 3.

Using the original *Monte Carlo* algorithm and evaluating each node as *success* or *failure*, being *success* pocketing the target ball in the target pocket and *failure* otherwise, we end with a sequence of shots rated with their probability of success. This approach highly depends on sampling and has a huge explosion of states generated since *Monte Carlo* performs a new search for every sample. For that reason, some pruning techniques were used. The easiest and more obvious technique is to stop the search when the probability of success of a given shot is under a certain threshold. Based on the results in [17, 18], shots besides the direct ones are only used 6% of the time. By generating shots for a given state of the table, one type at a time, in order of increasing difficulty, we can reduce even more the number of states generated when a direct shot is found.

Before the search starts, shots are rated and sorted based on their difficulty using a function from [8, 9]. This function is based on the distance between the balls and the pocket, and the cut angle, according to Equation 1.

$$\Delta = \frac{d_{co}d_{op}}{\cos^2\alpha} \quad (1)$$

\triangle is the shot difficulty, d_{co} is the distance between the cue ball and the target ball, d_{op} is the distance between the target ball and the pocket and α is the cut angle between the two balls. Since this formula penalizes long shots and high cut angles, the shots that will be explored first will be the ones closer to the cue ball and less vulnerable to noise. This formula does not need to be very precise since it will only serve to differentiate the easiest shots from the others. By rating all the shots with this formula and sorting the list in crescent order, we are guaranteeing that the shots with the highest probability of success (less difficult) will be at the beginning. Since we are evaluating one type of shot at a time, we stop the search when the difficulty of a shot is too high and give an opportunity to the next shot type, trying to find an easier solution.

Although we have significantly reduced the state space with these modifications, there is still the case where all the shots are too difficult and the cut condition is never reached. PoolMaster optimizes the shot parameters for every next ball with a local optimization algorithm, which is the approach that achieves better results. Therefore, if we use the search only to find the next ball and the sampling only to evaluate the probability of pocketing and repositioning, we will not need to perform a search in every sample. We already know which ball we want to reach and the probability of succeeding the plan will already give us the difficulty of succeeding. The question is: which table state should we use to search for the next ball? In this case, we used the noiseless state, since it will be the average resulting position of the cue ball.

By using the reposition for the next ball as part of the evaluation, we are forcing a shot to be at a good reposition. This allows us, in an implicit way, to benefit shots that not only break clusters, but also to guarantee that the cue ball will be at the best position possible for the next ball.

Shot Generator

The shot generator is responsible for generating all the possible shots for a given table state. In this work a backtracking search algorithm with ray tracing was used, this kind of search for shots guarantees from the root that the shots will be executable, and will also give us for free a complete description of what will happen on the table, such as the balls and rails involved and distance traveled. This information is very important to control the complexity of the shots being generated and also gives us a tool to control the skill on the player in terms of tactic behavior for a single shot.

The general algorithm is the following:

1. For every pocket, set the objective point as its center:
2. Cast a ray from every ball to the objective point:
 - (a) If the ray reaches the objective point, set the objective point as the center of the ghost ball position for this ball.
 - i. If the cut angle is greater than a certain limit, stop iteration on this path.
 - ii. Else, if this ball is the cue ball, calculate the shot parameters and add it to the list of shots.

iii. Else, go to step 2.

(b) Else, stop iteration on this path.

Ball collisions with high cut angles are not explored to remove shots that barely touch the balls and that do not produce relevant results.

For the case of the bank and kick shots, we used the PoolMaster table mirroring method adapted for ray tracing approach. For every rail collision allowed, a level of mirrors is added to the raytracer object list. For example, with 1 rail allowed, we add 4 mirrored tables (one on the left, right, top and bottom of the original table); with 2 rails allowed, we add 12 tables (4 from the level 1 and 8 around level 1 for the level 2). With this information on the raytracer, we can treat bank and kick shots like direct and combinations shots. However, the higher the number of rails allowed, the slower the ray tracing will be due to the number of objects in the list; that is why the bank and kick shots are only explored after combination shots. When using this method, we are assuming that the angle of incidence is equal to the angle of reflection, which is not true in the *FastFiz* engine.

Using the proposed approach, by only controlling the depth of the search and the number of rail collisions allowed, we have a general algorithm for almost every shot type without having to explicitly look for it.

The calculation of the initial shot parameters is done as follows:

- θ is set to the minimum possible value.
- ϕ is calculated aiming the cue ball center to the ghost ball center (the objective point of the ball before the cue ball).
- a and b are set to zero.
- V is retrieved from a precomputed table of minimum velocities.

To quickly find the minimum velocity required to pocket a ball, for a given shot situation, a minimum velocities table is precomputed. PickPocket [17, 18] and CueCard [2, 1, 4] generated direct shot situations by discretizing the cut angle (the angle that the cue ball makes with the target ball), the distance between the cue ball and the target ball, and the distance of the target ball to the pocket and simulating shots incrementing the velocity by 0.1m/s until the ball is pocketed. We generalized this to every shot type by discretizing the distance traveled by the sum of all the balls involved and the number of balls and rail collisions involved in the shot also.

At this point, we have a list of shots that put the target ball in the target pocket with the minimum velocity. There is an infinite number of variants of these shots that could still pocket the target ball. For position play, it is important to generate a set of shots that captures the range of possible follow-up states. The solution used to find the most significant variants was to pick n values equally spaced starting from the minimum parameter value up to the maximum for each shot parameter. The shots that accomplish the goal of pocketing the target ball are added to the shot list. Since the number of variants has a

huge impact on the branch factor of the algorithm, a study of the most relevant parameters was made in Section 5.

Evaluation Function

The evaluation function is responsible for differentiating shots from each other with a specific metric. In MiniPool, the evaluation of a shot is made by counting the number of times a shot is successful while sampling it a number of times with noise. A shot is considered successful if it pockets the target ball in the target pocket and it has a clear way to the target reposition point. If, while calculating this probability of success, a shot cannot reach a minimum threshold of probability, the evaluation stops. This is done to reduce the computation time on poorly reliable shots.

With this evaluation function, we have a metric of how difficult it will be to execute a shot that leads to a good reposition for the next one. There is no need for anything else since, if a shot is more successful than another, it is because it will be less vulnerable to noise. However, this approach might make a ball closer to a pocket better than another. According to Jack Koehler [12] these balls should only be pocketed in special situations. Foreseeing these situations requires a better plan for a sequence of shots which, due to shot execution time constraints, could not be done in this work.

RESULTS

To demonstrate the quality and potential of the approach developed, the results of various tests are presented in this section as well as the environment in which the tests were made. In all the tests only one component is modified in order to better demonstrate the impact of it. The graphics in every test show the accumulated average of the clean success percentage of the table (to demonstrate that it stabilizes before the end of the test), the time per shot and the reason why the algorithm stopped the iteration (to understand what is causing it to stop). The clean success metric is the number of times the player was able to pocket all the balls without losing the turn divided by the number of games played. The time per shot metric is the time passed since the beginning of the turn until execute the shot (computation time).

The tests were made using the *FastFiz* engine. For each test, 500 table configurations were randomly generated and the algorithm was executed until it loses the turn. The average time until it executes a shot and the reason why the iteration stopped are stored for each table. The computer used for the tests has a Windows 10 Pro Operating System, Intel Core i5 CPU at 2.30 GHz and 4 GB of RAM.

In *FastFiz*, shots are affected by a perturbation model, noise. The standard deviation values for each parameter are: $\phi = 0.125^\circ$, $\theta = 0.1^\circ$, $V = 0.075$ m/s, $a = 0.5$ mm and $b = 0.5$ mm. In these tests, the simulations were made at 0x and 0.5x of these deviations.

The maximum cut angle is set to 70° , the maximum number of balls and rails involved in the shot generation is set to 3 and 1 respectively, the maximum shot difficulty is set to 0.7, the minimum success probability is set to 60% and the acceptable probability of success is set to 80%.

Analysis

Results with noise

In general, the clean success probability in a noisy environment is very low comparing with the other players. PickPocket is able to reach 67% within 60 seconds per shot, JPool reaches 74% with 44 seconds per shot and PoolMasters reaches 97% with 19 seconds per shot. Looking at Figure 6, we can see that the main problem causing the low results are shots that failed to pocket the target ball. This problem can occur for two reasons; either the shot parameters were wrong or the shot was risky. This test was made with a sample size of 25, a depth of 2 and 125 variants of V , ϕ and b (5 per parameter). Looking at the results in Figure 8, by using a bigger sample size, we can increase by 10% the clean success probability, taking 7 more seconds per shot. In Figure 7 we used 2 more variants per parameter, resulting in 343 variants per shot. As expected, this did not change significantly the success of the player. For the change in the number of variants to be relevant, the gap between each parameter variant needed to be as small as possible, however for this gap to be small enough we would need much more than 7 variants per parameter. A possible solution in order to not depend on shot variants would be an optimization algorithm like PoolMaster did. For each ball-pocket combination it makes an optimization search to find the parameters that pocket the ball and reach a good position. This kind of approach however, relies on an objective function that needs to have a finer discretization for the algorithm to find a result faster. Finding such function in 8-Ball domain it is not an easy problem.

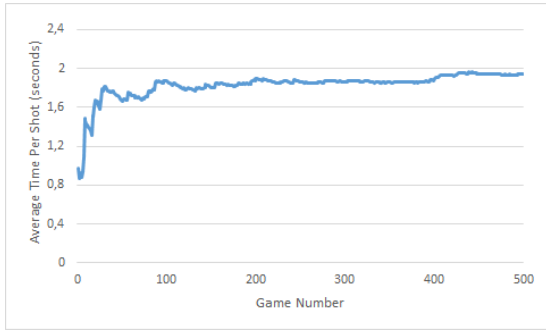
Results without noise

The results without noise were good comparing to others: JPool and PoolMaster are able to achieve 100% in approximately 44 and 19 seconds respectively. With a depth of 2 and 125 variants, MiniPool can reach clean table success of 86% in less than half a second. Using all shot types, we can have an improvement of 5% with a cost of 4 more seconds.

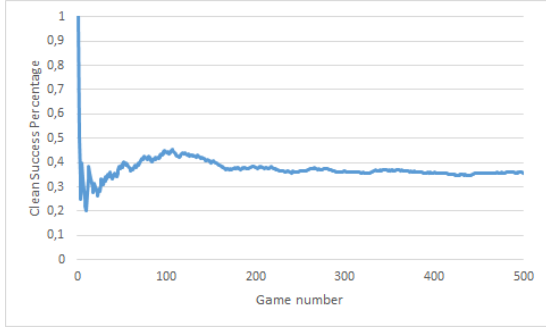
Since there is no noise in these tests, the reason for the algorithm to not be able to clean the table is probably a planning problem. To check this, a test was made using a depth of 3 (see Figure 11), however the results did not improve. Given that the results in Figure 10 were better, this drives us to conclude that there are situations when the algorithm is not able to continue because the target ball or pocket is obstructed. This situation prevents a direct shot from being executed for that ball. Since the algorithm searches for one type at a time, another direct shot will be chosen, and the next sequence of shots might put the cue ball in a situation where the algorithm cannot place it near the problematic ball again or reach that pocket. When we generate all shots at once, the algorithm will probably find a situation where the ball can be pocketed using another type of shot and solve the problematic ball.

CONCLUSION

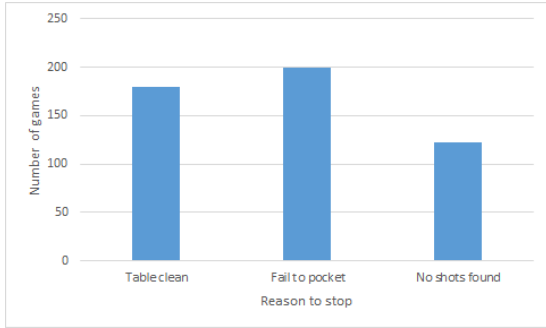
The main purpose of this work was to develop an artificial player for 8-Ball video game with a real-time response. Looking at the results of the tests without noise, we consider that this goal was achieved since videogames normally do not have noise perturbations. By ordering the shots by difficulty, taking



(a) Average time per shot stabilizes at 1,94 seconds



(b) Table clean success percentage, Stabilizes at 35.8%



(c) Motive for the algorithm to stop the iteration

Figure 6: Test results for 25 samples with 0.5x noise

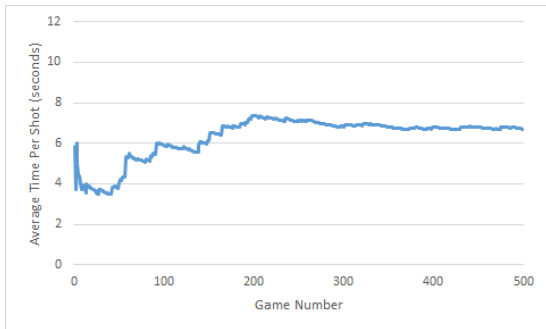
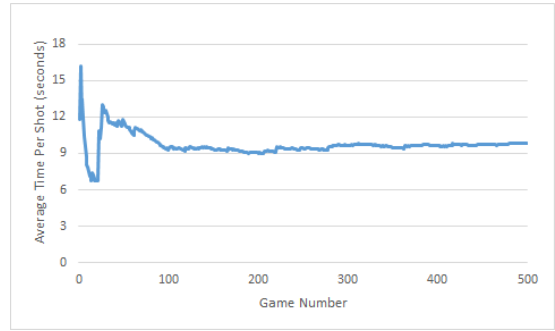
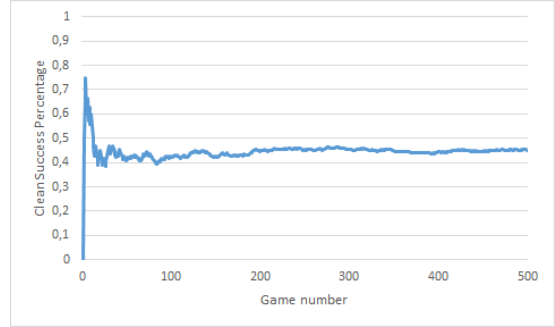


Figure 7: Average time per shot using 343 variants per shot. Stabilizes at 6.69 seconds with 0.5x noise

into account the distance of the balls, we are able to clear the table by zones. Combining this with the evaluation function, which benefits shots that are in a good position for the next ball, MiniPool can clear almost every table in less than half a second without having to search deeper in the tree.

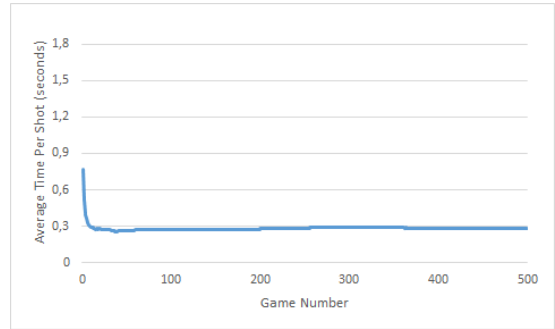


(a) Average time per shot stabilizes at 9,74 seconds

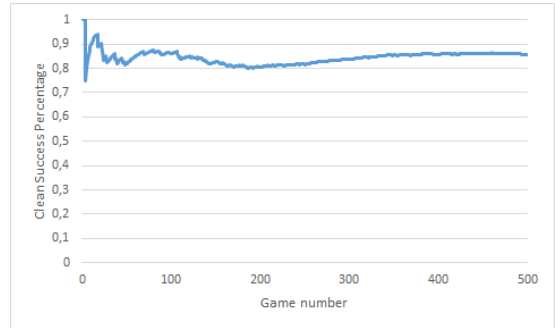


(b) Table clean success percentage, Stabilizes at 45.2%

Figure 8: Test results for 100 samples with 0.5x noise



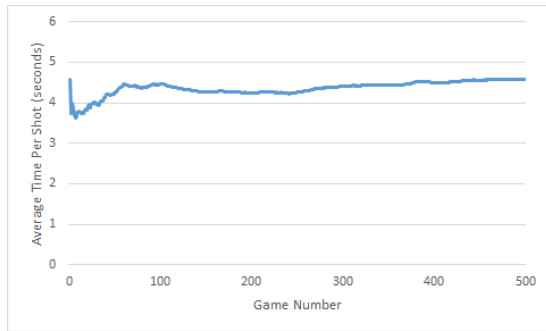
(a) Average time per shot stabilizes at 0.29 seconds



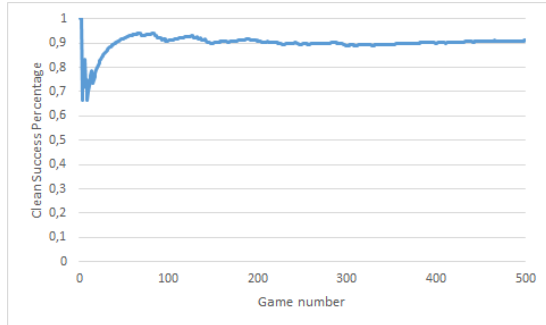
(b) Table clean success percentage, Stabilizes at 86%

Figure 9: Test generating shot types one by one without noise

On the other hand, a goal that was also in mind when developing MiniPool was to develop a player that could play in a environment with noise. The results for this case were not as good as expected and there are still some improvements that

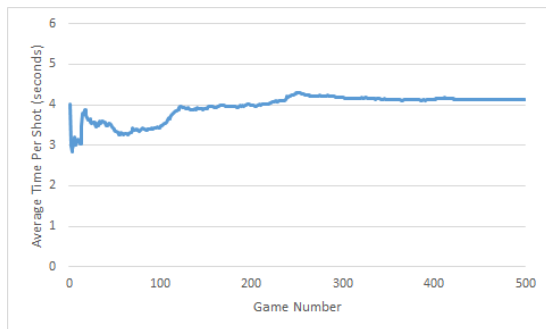


(a) Average time per shot stabilizes at 4.57 seconds

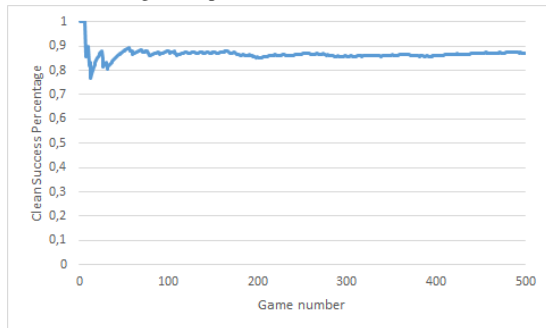


(b) Table clean success percentage, Stabilizes at 91%

Figure 10: Test generating all shot types without noise



(a) Average time per shot stabilizes at 4.12 seconds



(b) Table clean success percentage, Stabilizes at 87%

Figure 11: Test with a depth of 3 without noise

can be done as future work. One of the main problems of the algorithm was relying too much on the number of shot variants for the look-ahead. PoolMaster, by using the optimization approach, removed this dependence and could expend more

time generating a more robust shot. Using a similar approach in MiniPool might be the solution to improve the performance in a environment with noise.

Other improvements that can be done to reduce the time needed to generate the shots for a given state are the ray-tracing acceleration techniques, such as kd-trees. By using these techniques we can reduce the number of objects that need to be tested for collision, and optimize the performance of the raytracer. This optimization will probably allow us to generate all shot types at once with a lower cost in time.

MiniPool was developed to be highly configurable and give a complete control of its skill. Since MiniPool was developed to be used as an artificial opponent in an 8-Ball video game, it would also be interesting to study how to simulate several types of skill or even automatically adapt the skill to its opponent.

ACKNOWLEDGMENTS

Thank you Vânia Mendonça, Soraia Meneses Alarcão, Fabio Alves, Élvio Abreu, Ruben Rebelo, João Moreira, Luis Sampaio, Liliana Santos, Joaquim Silva, Simone Silva, Beatriz Branco, Conceição and Carla Branco for your feedback and support.

This work was supported by national funds provided through Fundação para a Ciência e a Tecnologia (FCT) (UID/CEC/50021/2013).

REFERENCES

1. Christopher Archibald, Alon Altman, Michael Greenspan, and Yoav Shoham. 2010. Computational pool: A new challenge for game theory pragmatics. *AI Magazine* 31, 4 (2010), 33–41.
2. Christopher Archibald, Alon Altman, and Yoav Shoham. 2009. Analysis of a Winning Computational Billiards Player.. In *IJCAI*, Vol. 9. Citeseer, 1377–1382.
3. Christopher Archibald and Yoav Shoham. 2009. Modeling billiards games. In *Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems-Volume 1*. International Foundation for Autonomous Agents and Multiagent Systems, 193–199.
4. Christopher James Archibald, Yoav Shoham, Jurij Leskovec, and Robert Wilson. 2011. *Skill and Billiards*. Stanford University.
5. Jens-Uwe Bahr. 2012. A computer player for billiards based on artificial intelligence techniques. (2012).
6. Nicolas Bureau. 2012. Sensibilité des coups au billard. *CaMUS, Université de Sherbrooke, Sherbrooke, QC, Canada* (2012), 9–26.
7. Subir K Chakrabarti. 2003. Pure strategy Markov equilibrium in stochastic games with a continuum of players. *Journal of Mathematical Economics* 39, 7 (2003), 693–724.
8. Shing Chyi Chua, Eng Kiong Wong, and Voon Chet Koo. 2005. Intelligent pool decision system using zero-order

- sugeno fuzzy system. *Journal of Intelligent and Robotic Systems* 44, 2 (2005), 161–186.
9. Shing Chyi Chua, Eng Kiong Wong, and Voon Chet Koo. 2007. Performance evaluation of fuzzy-based decision system for pool. *Applied Soft Computing* 7, 1 (2007), 411–424.
 10. Jean-Pierre Dussault and Jean-François Landry. 2005. Optimization of a billiard player–position play. In *Advances in Computer Games*. Springer, 263–272.
 11. Jean-Pierre Dussault and Jean-François Landry. 2006. Optimization of a billiard player–tactical play. In *International Conference on Computers and Games*. Springer, 256–270.
 12. Jack H Koehler. 1995. *The Science of Pocket Billiards*. Sportology publications.
 13. Jean-François Landry and Jean-Pierre Dussault. 2007. AI optimization of a billiard player. *Journal of Intelligent and Robotic Systems* 50, 4 (2007), 399–417.
 14. Jean-François Landry, Jean-Pierre Dussault, and Philippe Mahey. 2012. A robust controller for a two-layered approach applied to the game of billiards. *Entertainment Computing* 3, 3 (2012), 59–70.
 15. Jean-François Landry, Jean-Pierre Dussault, and Philippe Mahey. 2013. A heuristic-based planner and improved controller for a two-layered approach for the game of billiards. *IEEE Transactions on Computational Intelligence and AI in Games* 5, 4 (2013), 325–336.
 16. James MacQueen and others. 1967. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, Vol. 1. Oakland, CA, USA., 281–297.
 17. Michael Smith. 2006. Pickpocket: an artificial intelligence for computer billiards. In *Masters Abstracts International*, Vol. 45.
 18. Michael Smith. 2007. PickPocket: A computer billiards shark. *Artificial Intelligence* 171, 16 (2007), 1069–1091.

“So medieval like, so gentle, so perfect”: as categorias musicais da banda-sonora do videojogo *The Elder Scrolls IV: Oblivion*

Joana Freitas
CESEM – FCSH/NOVA
Lisboa, Portugal
joanasofsky@hotmail.com

ABSTRACT

A música para videojogos tem sido, cada vez mais, posta em destaque em diversos espaços e plataformas *online*, sendo alvo de muita actividade e produção com diversos propósitos por parte de utilizadores (Miller: 2012). É possível observar, especialmente no Youtube, a criação de listas de reprodução por utilizadores organizadas por ambientes, carácter e outros termos associados, como “épica”, “instrumental”, “atmosférica”, etc., cruzando uma série de bandas-sonoras de jogos.

Neste trabalho, é observada a forma como a música é sistematizada e inserida em categorias definidas de acordo com os diversos ambientes propostos no jogo *The Elder Scrolls IV: Oblivion* ao explorar diversos recursos musicais que são utilizados em função da nomenclatura aplicada a cada faixa, especialmente o ritmo, o timbre, andamento e instrumentação. Verificam-se os imaginários presentes no jogo, a sua construção e como é que estes são reforçados através da banda-sonora que acompanha a narrativa e experiência pessoal do jogador durante a jogabilidade, assim como esta pode ser considerada um depósito de significados associados às acções, atmosferas e contextos propostos ao utilizador.

Palavras-chave

Videojogos; *Oblivion*; música; ergodicidade; categorias; tópicos

INTRODUÇÃO

A banda-sonora de três títulos da série de videojogos *The Elder Scrolls* composta por Jeremy Soule foi colocada em destaque, entre muitos outros aspectos e em diversas plataformas, pelo site da estação de rádio britânica *Classic FM* em 2013 através do inquérito *online* realizado anualmente aos utilizadores/ouvintes da rádio de música clássica, o *Classic FM Hall of Fame* [1]. Nesse mesmo ano, o *Concerto para piano n.º 2* de S. Rachmaninov continuava em primeiro lugar pelo terceiro ano consecutivo, enquanto o *Concerto para piano n.º 5* “O Imperador” de L. V. Beethoven passou para sexto lugar devido à subida da música de J. Soule dos jogos *Elder Scrolls* para o quinto [2]. Mesmo não sendo um dos pontos mais relevantes no contexto deste trabalho, é curioso notar o facto desta votação ter tido reacções de diversas naturezas que se

centram principalmente na aproximação da música para videojogos à música clássica, e de como esta segue e realiza cruzamentos com outras formas de *media*, como o cinema e a televisão.

É possível verificar em muitos comentários de utilizadores – tanto no site da *Classical FM* como em vídeos de compilações de música para videojogos no Youtube –, para além de argumentar o seu “valor” enquanto “música”, referências à importância da música nos jogos e nos componentes que os constituem (gráficos e aspectos visuais, ambiente, interactividade, etc.) e a sua semelhança com outros meios audiovisuais, principalmente os filmes:

“I agree that if Beethoven were alive, he would be composing for movies - and perhaps even games too. Games have become increasingly cinematic and complex, and you spend upwards of 50 hours on many of them. For many game fans, the music is one of the most important parts of a game. Music draws you into that world, the same way it would for a movie, concert or symphony. Music is an experience of the senses; it is pretentious and erroneous to think that music is somehow less if it is accompanied with a visual or interactive component. For the first time in a long time you have flocks of young people voting and getting interested in classical music - hell yeah!”[3].

De acordo com Iain Hart, a musicologia dos videojogos (ou ludomusicologia [4]) tem sido construída e assente em três modos de questionamento: (1) o estudo da história do áudio e música em videojogos; (2) a relação e envolvimento com os diversos meios tecnológicos, bem como as suas inovações, na música de videojogos; (3) a análise de textos musicais de videojogos [5]. É importante referir que a música de videojogos se assemelha e faz referências directas a modos de construção e produção de outros meios audiovisuais, como já referido anteriormente; no entanto, este meio em específico exige do utilizador uma interacção mais activa e directa do que um ouvinte de música ou um espectador de um filme. Para além da interacção física através de um meio de *hardware* (um comando de consola ou um rato e teclado) que permite a troca de informação e o jogo, os videojogos exigem um modo de participação a nível intelectual, sendo necessário por parte do utilizador a realização de escolhas e decisões baseadas nos conteúdos

que o jogo providencia, determinando os diversos resultados que podem ocorrer e que moldam a experiência pessoal de cada jogador.

A criação dos vários elementos que constituem um jogo – a narrativa, o universo, as personagens, os efeitos visuais e sonoros, a música, etc. – é um processo geralmente estático (são elementos escritos e aplicados no jogo, podendo ser retirados do seu contexto, em conjunto ou individualmente, e analisados de uma forma linear); contudo, a percepção destes elementos é realizada de uma forma dinâmica [5]. Cada jogador tem a sua própria percepção, moldada pela construção da sua identidade, diversos contextos sociais que se insere, entre muitos outros aspectos, pelo que a mensagem que os elementos do jogo constroem é distinta para quem joga, sujeita a constantes alterações. Ainda segundo Hart (um dos poucos autores que procura compreender a música de videojogos como um processo comunicativo de significado através de performance e interactividade), o áudio e a música criados para o jogo podem ser considerados como depósitos de significado dentro de um ambiente relativamente estático, no qual o jogador percebe-os como partes integrantes da sua experiência dinâmica, onde ambos o contexto e o significado se podem alterar.

Jeremy Soule refere em vários dos seus comentários partilhados nas suas próprias páginas em redes sociais (principalmente o *facebook* e *Twitter*) características, motivações e outras informações sobre as músicas que produz. Tomando como exemplo a citação seguinte, Soule discute uma faixa específica do videojogo *The Elder Scrolls V: Skyrim* (Bethesda Softworks, 2011) (doravante referido como *Skyrim*), denominada *Standing Stones* e que está inserida na categoria proposta pelos *designers* do jogo como “explore” (exploração) [6]:

“[T]his composition was challenging despite the slow tempo and monolithic movements. My multi-layered approach was to embolden the player with a narrative. This was accomplished through a specific structure. The first part of my structure/device was the acknowledgement of adversity. The second was to provide an enlightened incentive to overcome the adversity, and the third part of my construct was to provide a sense of scale of the world to create relevance for the entire idiom (the frame of the painting). Overall, I wanted the theme of the narrative to be about a dark hope and an illustration of an almost unsaid acknowledgement of a repressed truth. This narrative also starts with a metaphor. In this metaphor, I presented what I call the “walker” (or soul) whose footsteps are represented in plodding half notes. The next cast members of this metaphor include nature, light, darkness and death itself (adversity).”[7]

Tanto *Skyrim* como *The Elder Scrolls IV: Oblivion* (Bethesda Softworks, 2006) (doravante referido como *Oblivion*) têm, nos ficheiros de instalação do jogo presentes na plataforma de computador (PC), a pasta “Music”

subdividida em categorias de acordo com o local onde as músicas são usadas ou a sua função, desde “combat” a “town”. Cada conjunto de faixas engloba uma série de sonoridades, propostas de instrumentação e texturas que são, até a uma certa extensão, semelhantes, de forma a corresponder a imaginários presentes no universo dos jogos. Para além de um propósito primeiramente funcional dentro do *design* computacional para activar uma série de músicas em sequência de acordo com o local onde o jogador se encontra durante a sua experiência pessoal do jogo, cada música procura transmitir ao jogador, de forma consciente ou não, diversas associações culturais através de tópicos musicais, como os sopros de metal (frequentemente as trompas) em situações de batalha [8] – ambientes onde se promove a energia e acção por parte do jogador cujo personagem está em perigo – ou as “(...) plodding half notes” como os passos do “walker” [9] em *Standing Stones*.

Considerando o processo pelo qual a música é sistematizada e inserida em categorias definidas de acordo com os diversos ambientes propostos no jogo *Oblivion*, este trabalho observa que motivos, materiais, instrumentação e outros recursos musicais são utilizados em função da “etiqueta” que é aplicada a cada faixa e como é que esta estruturação é realizada. Procurando cruzar aspectos teóricos de autores como Monelle, Peirce e Aarseth, este trabalho propõe uma leitura pessoal sobre os imaginários presentes nos jogos – desde “medieval” a “fantástico” como “celta” e “viking” – e como é que estes são reforçados através da banda-sonora que acompanha a narrativa e experiência pessoal do jogador durante a jogabilidade. Ao perpetuar códigos musicais estereotipados presentes em meios audiovisuais já anteriores aos videojogos, esta banda-sonora pode também ser considerada um depósito de significados associados às acções, atmosferas e contextos propostos ao utilizador.

THE ELDER SCROLLS IV: OBLIVION: CONSIDERAÇÕES SOBRE A MÚSICA NO UNIVERSO DE CYRODIIL

O videojogo *Oblivion* é um RPG (*role-playing game*) passado na província de *Cyrodil* (a qual está inserida no universo de *Tamriel*) composta por nove cidades. Para além das missões (*quests*) que fazem parte da narrativa do jogo que o jogador pode executar, este título insere-se nos moldes de um jogo *open world*, no qual o jogador tem a possibilidade de explorar livremente o ambiente do mesmo, incluindo as cidades, aldeias, florestas, montanhas, caves, etc., nas quais habitam outros personagens (NPCs), animais – agressivos e/ou passivos – e criaturas, desde minotauros a fantasmas e *trolls*.

A banda-sonora (não diegética) de *Oblivion* tem aproximadamente 1 hora (59:30) [10], dividida em 28 faixas embora apenas 26 das mesmas é que estão incluídas [11]. No directório de instalação dos ficheiros do jogo no PC, a pasta “Music” está subdividida em cinco categorias,

nas quais cada conjunto de faixas tem um sistema-nominal para os ficheiros de acordo com a sua função, provavelmente para facilitar o processo de produção e de funcionamento das músicas *in-game*. As músicas têm ainda um título específico individualmente dado pelo compositor que é visível se o utilizador comprar o CD em suportes físico e/ou digital ou pesquisar *online*.

Ao estar presente numa zona florestal do jogo, por exemplo, e o jogador encontrar umas das criaturas previamente referidas, por exemplo, um *troll*, uma das músicas que se inserem na categoria “battle” vai interromper a música que estava previamente a acompanhar a acção, esta inserida na categoria “explore”.

Sendo a música uma parte integrante de um texto audiovisual, esta não é a única componente semiótica que opera nos videojogos – este meio consiste de múltiplos integrantes semióticos a actuar em conjunto. De acordo com Hart, a existência de um processo de significação, entre o jogador e o jogo indica que o conjunto dos vários componentes que o integra é, de um ponto de vista semiótico, maior do que as suas partes isoladas. A música que é ouvida ao jogar *Oblivion* é considerada pelo jogador a música “do *Oblivion*” em vez de a música que esse jogador ouviu enquanto estava a jogar simultaneamente *Oblivion* [5].

No livro *The Musical Topic: Hunt, Military and Pastoral*, Monelle propôs uma classificação de tópicos como icónicos ou indexicais (a partir das relações que Peirce discute entre ícone, índice e símbolo), de acordo com a sua representação tanto de sons naturais (como o cavalo ou o pássaro) ou de eventos musicais. O autor afirma que é necessária uma metodologia distinta ao estudar estes dois tipos de tópicos, pois estes têm características diferentes: o tópico icónico é musical apenas no seu significante, sendo o significado natural, social ou histórico [8]. Monelle exemplifica com o cavalo, em que o objecto do tópico “cavalo nobre” ou o som do galope do mesmo não são musicais, apenas o seu significador – como, por exemplo, o ritmo ou compasso 6/8, são musicais. Este objecto é referido como *fundamento* – segundo Peirce, o signo representa alguma coisa, o seu Objecto [12] – e ele representa-o em referência a uma espécie de ideia, o que pode ser denominado como *fundamento*). O autor reforça ainda a ideia de que o tópico indexical, tal como o icónico, tem um mundo de diversos significados associados para além do repertório musical, como na literatura, relacionadas com caracteres morais, como heróico ou aventureiro. Este tópico requer a descrição do próprio repertório, do fenómeno musical, aspectos organológicos, etc., devido ao lado musical do seu significador, tomando como um dos exemplos possíveis o tópico da chamada militar da trompete remetendo para as chamadas realizadas por estes instrumentos pelos exércitos já no séc. XIX.

Monelle afirma que a diferença principal ao estudar tópicos indexicais e tópicos icónicos é a importância do

fundamento: o tópico indexical é observado relativamente ao nível de aproximação entre o timbre, ritmo, motivos melódicos, etc. da música ao repertório ecoado; enquanto o tópico icónico não tem uma música original, mas sim um conteúdo musical que é uma imitação da natureza.

Entre as várias músicas inseridas na categoria “battle”, pode-se retirar como exemplo a música *Daedra In Flight*. Reunindo algumas informações de uma breve análise auditiva, esta faixa apresenta características como:

- Um compasso composto;
- Ostinato rítmico sincopado nas cordas num registo grave;
- Percussão que marca o tempo forte e acompanha o acelerando, incluindo um sino;
- Crescendo progressivo em camadas nos sopros de metal;
- Um elemento percussivo (pandeireta ou guizos) antecedente da entrada do flautim com um novo motivo melódico-rítmico;
- A transição realizada pelo flautim para um outro motivo nos sopros de metal (trompas) com figuração rítmica longa e intervalos ascendentes.



Figura 1: Transcrição pessoal do excerto inicial da música *Daedra in Flight*, motivo/ostinato rítmico

Monelle aponta para a complexidade do tópico militar e as questões históricas em volta das próprias práticas musicais em contextos bélicos: “*The contour of the military topic is different from that of the hunt. One cannot trace offshoots and developments, as one can for the hunt topic. There is nothing like the pastoral horn or the “horn of nocturnal mystery”*”.

On the other hand, both march and trumpet signal have varied backgrounds, and although the warlike flavor is usually apparent, there is always a hint of the ceremonial, of civic splendor as well as soldiering.” [8]

A nível musical, estas características podem ser referências indexicais a estes imaginários, evocando universos bélicos, militares, entre outros, reforçadas por um ritmo dactílico que remete para a cavalgada. Pode ainda referir do mesmo modo o cavaleiro nobre das batalhas na Idade Média, e, em



Figura 2: Transcrição pessoal do motivo presente na entrada do flautim na música *Daedra in Flight*



Figura 3: Transcrição pessoal do motivo presente na(s) entrada(s) da(s) trompa(s) na música *Daedra in Flight*

Não é necessário, nem talvez pertinente, observar todas as músicas desta categoria, tendo em conta que, na sua totalidade, as várias faixas partilham características entre si como as já referidas relativamente a *Daedra In Flight*, principalmente andamentos acelerados com ritmos marcados e linhas melódicas em destaque em instrumentos de sopro, madeiras e/ou metais.

Um outro aspecto interessante a discutir neste contexto é a questão da interpretação ergódica por parte do jogador e a sua (possível) relação com a música. De acordo com a teoria cibertextual proposta por Espen Aarseth, o cibertexto tem de conter informação que funcione de uma forma recíproca e em *loop*, que se traduz em percepções distintas cada vez que o utilizador interage e interpreta o conteúdo (como um texto lido diversas vezes). Segundo Aarseth, a cibertextualidade não é um novo modelo de texto mas sim uma perspectiva nas múltiplas formas textuais de forma a expandir o âmbito dos estudos literários para incluir fenómenos que, à partida, seriam marginalizados pelo

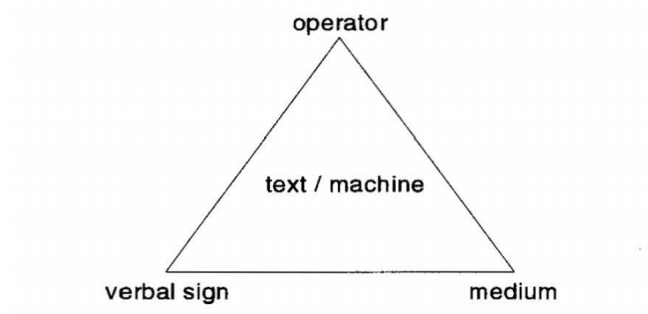


Figura 4: *The Textual Machine* (Aarseth: 1997)

mesmo campo. É importante referir ainda que:

“[A]s the cyber prefix indicates, the text is seen as a machine-not metaphorically but as a mechanical device for the production and consumption of verbal-signs. Just as a film is useless without a projector and a screen, so a text must consist of a material medium as well as a collection of words. The machine, of course, is not complete without a third party, the (human) operator, and it is within this triad that the text takes place. (See figure 1.1.) The boundaries between these three elements are not clear but fluid and transgressive, and each part can be defined only in terms of the other two. Furthermore, the functional possibilities of each element combine with those of the two others to produce a large number of actual text types.” [13]

O autor analisa ainda jogos digitais de aventura da década de 90 como um objecto de interpretação ergódica, contendo elementos dinâmicos e colocando a questão da

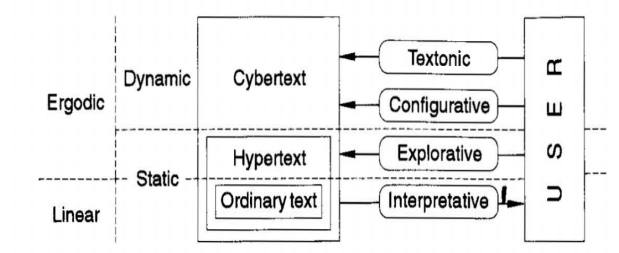


Figura 5: User Functions and Their Relations to Other Concepts (Aarseth: 1997)

interactividade em destaque, afirmando que estas formas de texto exigem um esforço não-trivial por parte dos utilizadores, os quais participam na construção dos mesmos. [13]

Por outro lado, James Newman discute o “mito” dos videojogos ergódicos, propondo que este meio não é interactivo nem ergódico, e que não providencia apenas um tipo de experiência e um modo de envolvimento. O autor afirma que pode ser visto como um depósito de elementos ergódicos ou interactivos, constituído por estruturas e segmentos muito definidos que formam secções com bastante ergodicidade, e controlado por sequências cinemáticas, mapas, sistemas de pontuação, entre outros. [14]

Em vários jogos, é apresentado ao jogador uma série de diversas informações, tendo como objectivo, por exemplo, a contextualização da narrativa dos mesmos, os seus objectivos, etc., sob a forma de uma cinemática inicial, por exemplo, o que não permite a interacção física do utilizador com o jogo (excepto nos casos que se pode premir uma tecla específica para passar para a secção seguinte). Dependendo da construção do arco narrativo do jogo, estes podem ser constituídos por muitos segmentos alternados ergódicos e “não-ergódicos”, formando uma experiência incontinua de jogabilidade.

No caso de *Oblivion*, o jogador é confrontado com uma cinemática inicial a apresentar o ponto de situação da narrativa através da personagem representativa do Imperador de *Cyrodill*. Este tipo de mecânica visual é utilizado apenas neste jogo em situações muito específicas, nomeadamente a inicial, no momento de batalha final e no final da narrativa linear, o que não impede ao jogador de continuar a explorar o mundo de *Oblivion*. Removendo destas observações os sinais e mensagens que surgem no jogo quando uma missão é aceite ou um objecto importante é encontrado ou recebido pela personagem que o utilizador controla, a sua experiência de jogabilidade não é interrompida por estes elementos visuais, seja ao explorar uma gruta ou a descansar numa taberna. Um elemento que participa na interrupção da percepção ergódica do contexto em que o jogador se insere é a música, principalmente a da categoria “battle”, devido às diferenças de timbre, andamento e instrumentação bastante contrastantes relativamente às restantes categorias.

Vários códigos musicais estabelecidos em meios anteriores aos videojogos, principalmente no cinema e discutido por autores desde Theodor Adorno e Hans Eisler [15], Claudia Gorbman [16], Marcia Citron [17] ou James Buhler [18] atribuem valor à imagem ou reforçam-na através do som e música, cujos fenómenos associados também foram interpretados por Michel Chion [19]. O jogador pode estar numa destas áreas referidas – floresta, montanhas, cidades, etc. – e uma das músicas inseridas na categoria “battle” começar, sem estar no seu campo visual qualquer ameaça.

A música é activada simplesmente por estar uma ameaça directa à personagem num raio específico (podendo exemplificar com os casos de um guarda que descobriu um crime que o jogador efectuou com a sua personagem; animais selvagens como ursos ou lincos; e ainda seres fantásticos como fantasmas, *trolls*, etc.). Isto transmite directamente ao jogador que está em perigo; o contexto histórico do jogo em si e a estética aplicada – nomeadamente através referências medievais e o reforço da ideia de um herói que o jogador tem de ser para completar a missão principal do jogo (salvar *Cyrodill* da ameaça iminente que, como habitual, vem de uma outra dimensão maléfica, neste caso, o reino de *Oblivion*,) – é valorizado principalmente pelos motivos rítmicos e tímbricos, como a percussão e as trompas – que fazem referências, por vezes, à banda-sonora em contextos bélicos de Howard Shore em *Senhor dos Anéis* (P. Jackson, 2001, 2002, 2003) (universo que, por vezes, é uma clara referência e inspiração para a construção do ambiente deste jogo).

H. Shore associa as trompas não só à valorização da coragem dos humanos no reino de Gondor, como os ritmos sincopados nas cordas graves que efectuam motivos curtos e acentuados nas trompas para acompanhar a produção, treinos, marchas e a própria representação dos *Orcs* (considerados uma “raça básica e pouco civilizada” [21]). O facto de o jogador ter de procurar a origem visual da

ameaça e processar qual o tipo de interacção que irá desempenhar transforma o significado não só visual como também musical recebido pelo jogador, interpretando (ergodicamente) a ameaça e o acompanhamento musical a ela associado como uma oportunidade de ser – ou tentar – um “herói”, fugir, esconder-se, ou o que cada jogador decidir dentro das suas próprias opções.

Em tom de contraste, as categorias “explore” e “public” partilham algumas afinidades entre as músicas que as compreendem, principalmente a nível de instrumentação – dando destaque à flauta, cordas e, por vezes, harpas, tanto como numa linha principal como desempenhando a função de preenchimento harmónico – andamento e texturas. Faixas como *Sunrise of Flutes* ou *Harvest's Dawn* (da categoria “public”, ou seja, apenas quando o jogador se encontra em cidades) são maioritariamente compostas por dois a três motivos principais na flauta ou na harpa, com a melodia e ritmo bem definidos, que são sequenciados à medida que vão sendo repetido e explorados. As faixas da categoria “explore” são, na maior parte, um pouco mais longas do que as da categoria anterior, e distinguem-se principalmente devido à sua construção por camadas nos vários naipes instrumentais, assentando também em certos motivos específicos de cada faixa que são repetidos ao longo da música, intervalados por longas frases melódicas geralmente nas cordas.

Paul Martin analisa o universo e paisagens de *Oblivion* a partir de um quadro teórico formulado por Burke e Kant relativamente ao sublime e ao pastoral, propondo que, à medida que o jogador vai explorando este mundo e ficando cada vez mais familiarizado, a imensidão da paisagem apresentada inicialmente ao utilizador através da cinemática inicial – considerada sublime – é reduzida para uma percepção pitoresca por parte do observador que pode estar relacionada com a resolução final na narrativa do conflito central do jogo. O autor afirma que:

“The opening scene of Oblivion clearly establishes the 19th century pastoral design as described by Marx. The bucolic, pre-industrial repose of Tamriel is threatened by an industrial Oblivion. In the Oblivion version of the pastoral the encroaching forces serve as a call to arms. It is not the idyll that is the focus but rather the threat to it. The abiding energy is not one of repose or admiration but of what is anathema to the pastoral (and fundamental to video games): action. (...) In short, the awesome breadth of Tamriel has been transformed over the course of the game to a set of discrete, manageable spaces. Unlike in comparable epic films, in which each landscape shot presents as the sublime, once the player is allowed to walk the fields - and outside the sublimely framed shot - the sense of grandeur can no longer be sustained, or at least it can only be sustained fitfully.” [20]

De acordo com Monelle, há vários mundos de tópicos que constituem géneros musicais e culturais, incluindo o pastoral. O timbre da flauta e oboé ou a estrutura rítmica da

siciliana podem actuar como significantes, e a tranquilidade do pôr-do-sol ou paisagens pacíficas como significados [8]. Embora os “(...) musical signs are sonic events that create an effect in a perceiver; not everything happening in music necessarily functions as signs all the time (something might not be apprehended, might not cause an effect)” [22], é possível também associar os timbres explorados principalmente pela categoria “explore” e a sua presença no acompanhamento musical de contextos ambientais em que estão associados ao universo deste tópico, reforçando a ideia de tranquilidade ou de aventura em músicas como *Peace of Akatosh* ou *Glory of Cyrodiil*.

Os títulos dados às várias faixas são também um aspecto pertinente a verificar, ligando-se às relações transtextuais entre textos discutidas por G. Genette (as quais não são herméticas, reforçando a rede de interacções que se pode estabelecer entre os conjuntos de textos), e que se traduz nomeadamente na Paratextualidade [23]. Podendo consistir em títulos, subtítulos, epígrafos, notas de rodapé, etc., e, na música, as indicações interpretativas e expressivas, de andamento, bem como os títulos das peças, são elementos muito importantes na tentativa de controlo e/ou influência na recepção de um texto (ou obra musical) pelo utilizador: “*Seriously, music were so good on Oblivion that even their names sounds a serious classic deal. That's devotion right there*” [24].

Ao olhar para o título das faixas, é possível concluir através dos termos utilizados que tipo de imaginários cada música procura corresponder. Na categoria “battle”, observa-se a sua inserção em contextos bélicos através de referências à marcha em *March of the Maradeurs*, a armas em *Fall of the Hammer* e *Bloody Blades*, e violência em *Defending the Gate* ou *Bloodlust*. Já na categoria “dungeon” (da qual as músicas acompanham apenas quando o jogador se encontra em caves, grutas, etc., e são compostas, na sua maior parte, por cordas em registos médios a graves, sem uma estrutura tonal definida nem motivos perceptíveis), encontra-se uma faixa denominada de uma forma bastante directa como *Tension*, e outras com referências a objectos distantes e sombrios, como *Deep Waters* ou *Wind from the Depths*. As categorias “public” e “explore” já foram discutidas relativamente aos imaginários de tranquilidade e pastoral veiculados pelas faixas que as integram, reforçadas por títulos como *Harvest Dawn*, *Dusk at the Market* ou *All's Well*.

Categoria	Nome do ficheiro	Título	Duração
Explore	atmosphere_01.mp3	<i>Glory of Cyrodiil</i>	02:28
	atmosphere_03.mp3	<i>Through the Valleys</i>	04:19
	atmosphere_04.mp3	<i>Ministrel's Lament</i>	04:42
	atmosphere_06.mp3	<i>Auriel's Ascension</i>	03:05
	atmosphere_07.mp3	<i>Wings of Kynareth</i>	03:30

	atmosphere_08.mp3	<i>King and Country</i>	04:05
	atmosphere_09.mp3	<i>Peace of Akatosh</i>	04:11

Tabela 1: Categoria musical “Explore” e respectivas faixas musicais do jogo *Oblivion*

Categoria	Nome do ficheiro	Título	Duração
Special	success.mp3	N/A	00:17
	death.mp3	N/A	00:13

Tabela 2: Categoria musical “Special” e respectivas faixas musicais do jogo *Oblivion*

Categoria	Nome do ficheiro	Título	Duração
Public	town_01.mp3	<i>Harvest Dawn</i>	02:51
	town_02.mp3	<i>All's Well</i>	02:26
	town_03.mp3	<i>Sunrise of Flutes</i>	02:56
	town_04.mp3	<i>Watchman's Ease</i>	02:05
	town_05.mp3	<i>Dusk at the Market</i>	02:11

Tabela 3: Categoria musical “Public” e respectivas faixas musicais do jogo *Oblivion*

Categoria	Nome do ficheiro	Título	Duração
Battle	battle_01.mp3	<i>March of the Maradeurs</i>	02:08
	battle_02.mp3	<i>Bloodlust</i>	01:07
	battle_03.mp3	<i>Daedra in Flight</i>	01:02
	battle_04.mp3	<i>Fall of the Hammer</i>	01:16
	battle_05.mp3	<i>Bloody Blades</i>	01:14
	battle_06.mp3	<i>Defending the Gate</i>	01:26
	battle_07.mp3	<i>Death Knell</i>	01:07
	battle_08.mp3	<i>Churl's Revenge</i>	01:08

Tabela 4: Categoria musical “Battle” e respectivas faixas musicais do jogo *Oblivion*

Categoria	Nome do ficheiro	Título	Duração
Dungeon	dungeon_01_v2.mp3	<i>Wind from the Depths</i>	01:42
	dungeon_02.mp3	<i>Tension</i>	02:32

	dungeon_03.mp3	<i>Ancient Sorrow</i>	01:05
	dungeon_04.mp3	<i>Deep Waters</i>	01:11
	dungeon_05.mp3	<i>Unmarked Stone</i>	01:06

Tabela 5: Categoria musical “Dungeon” e respectivas faixas musicais do jogo *Oblivion*

OBSERVAÇÕES FINAIS

A presença do compositor nas redes sociais, nomeadamente no *facebook*, é útil, numa certa extensão, devido à publicação de comentários pessoais relativamente aos vídeos das músicas de jogos para os quais compôs e que J. Soule partilha no seu perfil. No caso de publicações relativas a *Oblivion*, J. Soule faz comentários a duas músicas em específico: *Fall of The Hammer* (uma das músicas na categoria “battle”) e *Ministrel’s Lament* (uma das músicas na categoria “explore”). Relativamente à primeira, Soule afirma que

“*Celtic dance music has always fascinated me, and it is an inspiration I incorporated into this melee composition. There’s just something that comes from nature itself when Celtic melodies and progressions are presented. And these sounds, once married into the ultimate human instrument known as the symphonic orchestra, tell an epic tale.*” [25]

Tendo estado encarregue da composição da banda-sonora para três instalações da série *Elder Scrolls – The Elder Scrolls III: Morrowind* (2000), *Oblivion* e *Skyrim*, é possível notar a presença de motivos partilhados entre algumas faixas de jogos distintos, como a referência de *Ministrel’s Lament* da categoria “explore” de *Oblivion* na música *Wind Guide You* da mesma categoria em *Skyrim* [26]. No entanto, o tipo de texturas e instrumentação entre estas duas últimas instalações é claramente mais distinto, transitando para uma orquestra com mais percussão, metais, e a inclusão de coros masculinos com textos na língua desenvolvida para o jogo (*Dovahzul* ou língua de dragões/“dragonês” [27]) para corresponder ao imaginário celta e viking que *Skyrim* procura representar, provavelmente inspirados ou pensados desde a composição de *Fall of the Hammer* para *Oblivion*. O ambiente frio e nórdico deste jogo, aliado ao contexto narrativo centrado no súbito regresso de dragões dos quais a personagem do jogador (que, como seria de esperar, é um novo herói chamado *Dragonborn*) tem de salvar a província de *Skyrim* (outra vez), promove a exploração de toda a paisagem e a sobrevivência em ambientes hostis do utilizador através de treino físico ou mágico, recolha de materiais e caça de animais para mantimentos. A presença das trompas principalmente em faixas para *dawn* e *dusk* pode aliar-se ao universo do tópico da caça, reforçando o “(...) heroic, virile, outdoor, adventurous character.” [8].

Este trabalho explorou brevemente que tipo de imaginários (visuais e musicais) são veiculados na banda-sonora de *Oblivion*, focando principalmente a categoria “battle” devido às suas características tímbricas e rítmicas distintas das outras categorias e que tem um papel talvez mais complexo na interpretação ergódica do cibertexto, neste caso, o videojogo. Seria interessante ainda, num contexto futuro, observar as principais diferenças e tópicos musicais explorados na banda-sonora de *Skyrim* e procurar compreender como é que, através dos mesmos sistemas compositivos e de organização de ficheiros musicais no sistema do jogo, a estética, sonoridade e

texturas apresentam-se distintas de uma forma muito clara, considerando que são características que a banda-sonora de J. Soule procura corresponder devido à proposta de um novo universo, mais “medieval” e “épico”, nesta instalação.

O acesso limitado – ou inexistente – a partituras e outras fontes musicais não permitiu uma análise mais extensa para além da empírica e auditiva, pelo que foi necessário recorrer ao *software* de edição musical *Finale 2014* para transcrever os motivos melódico-rítmicos em anexo como ilustração de algumas características a focar relativamente à faixa *Daedra In Flight*. É importante ainda mencionar que a escassez de literatura primária sobre semiótica da música em videojogos foi um outro aspecto que dificultou a concepção de um quadro teórico mais definido, resultando no cruzamento entre referências inseridas em *game studies* e a musicologia nos *new media*.

Contudo, a actividade constante e interacção *online* entre utilizadores, neste caso, sobre os videojogos e, em particular, sobre a música, permite observar discursos, reflexões e até práticas em torno deste fenómeno (ainda que seja considerado, por vezes, discreto na indústria e na academia) extremamente variados e cada vez mais frequentes. *Sites*, plataformas, redes sociais, fóruns, etc., com as múltiplas cibercomunidades que albergam e que operam nestes meios, podem ser considerados fontes e recursos primários para a pesquisa e metodologia de trabalhos científicos na área de *media*, especificamente nos videojogos. Assim, é muito possível verificar a existência de informações, comentários e percepções por parte de utilizadores relativamente a este meio e inclusive a música que o acompanha, frequentemente associada a emoções e estereótipos, desde a “nostalgia” associada a jogos das décadas de 80 e 90, como a “grandiosidade” da banda-sonora de *Skyrim* ou os toques “medievais” na música de *Oblivion*, permitindo múltiplas abordagens teóricas e interdisciplinares não só da música e universos sonoros dos videojogos, como do próprio meio.

REFERÊNCIAS

1. A descrição do *site* apela e reforça a acção do utilizador no seu voto, de modo a relevar “the ultimate chart of classical works”. O guia do último ano, 2015, e restantes informações podem ser acedidos aqui: <http://halloffame.classicfm.com/2015/> (último acesso a 28 de Setembro 2016)
2. A música para a série de videojogos *Final Fantasy* do compositor Nobuo Uematsu também é de destacar, tendo ficado em terceiro lugar na votação. <http://www.classicfm.com/hall-of-fame/2013/video-games-beethoven-hall-fame-top-5/> (último acesso a 28 de Setembro 2016).
3. Comentário publicado pelo utilizador Siobhan Bryn na secção de comentários em resposta à proposta de discussão mencionada no artigo aqui: <http://www.classicfm.com/hall-of-fame/2013/video-game-music-hall-fame-opinion/#qxLEMJCgyh5QmTJp.97> (último acesso a 28 de Setembro 2016)
4. O termo ludomusicologia está directamente associado ao cruzamento da musicologia e ludologia, principalmente focado no estudo da música de videojogos, ganhando cada vez mais relevo na academia, tendo sido já fundado em 2011 um grupo de investigação homónimo do Reino Unido. Para mais informações sobre as primeiras utilizações deste termo neste contexto, ver: Moseley, R. *Playing Games with Music (and Vice Versa): Ludomusicological Perspectives on Guitar Hero and Rock Band* (2012), http://www.rogermoseley.com/Music/musicology/Entries/2012/1/19_Playing_Games_with_Music_and_Vice_Versa_Performance_and_Recreation_in_Guitar_Hero_and_Rock_Band_files/Playing%20Games%20with%20Music.pdf (último acesso a 28 de Setembro 2016).
5. Hart, I. Meaningful Play: Performativity, Interactivity and Semiotics in Video Game Music. *Musicology Australia* (2014), 36:2, 273-290
6. Embora este comentário seja relativo à quinta instalação da série *Elder Scrolls*, a qual não será analisada neste trabalho, é pertinente observar não só como o compositor discute os seus próprios processos de composição e interacção com a narrativa, como estes se assemelham aos métodos compositivos utilizados na instalação anterior, *Oblivion*, o objecto de estudo aqui em foco.
7. Comentário do compositor Jeremy Soule na música *Standing Stones* da banda-sonora do videojogo *The Elder Scrolls V: Skyrim* (2011, Bethesda Softwork) publicado no *facebook* pessoal (<https://www.facebook.com/OfficialJeremySoule/posts/375979709147161>) e referenciado na *wiki* da série *Elder Scrolls*, que pode ser acedido aqui: http://www.uesp.net/wiki/Skyrim:Music#cite_note-JSStandingStones-4 (último acesso a 30 de Abril 2016)
8. Monelle, R. *The Musical Topic: Hunt, Military and Pastoral*. Indiana University Press (2006)
9. Embora esta referência que o compositor faz não seja muito clara, é provável que o “walker” (ou “soul”) seja o personagem do próprio jogador.
10. A banda-sonora completa pode ser ouvida aqui: <https://www.youtube.com/watch?v=SpqSdORmCX4>
11. Duas das faixas são de uma duração muito curta – menos de 15 segundos – que surgem quando o personagem do jogador sobe de nível ou morre, respectivamente. Verificar as tabelas 1, 2, 3, 4 e 5 de todas as faixas de acordo com a sua categoria. Todas as informações foram consultadas aqui: <http://www.uesp.net/wiki/Oblivion:Music> (último acesso a 28 de Setembro 2016)
12. Peirce, C. S. *The philosophy of Peirce: selected writings*. Londres, Kegan Paul (1940), 102-107
13. Aarseth, E. *Perspectives on Ergodic Literature*. Johns Hopkins University Press (1997). Os esquema referidos no texto encontram-se nas Figura 4 e 5.
14. Newman, J. The Myth of the Ergodic Videogame: Some thoughts on player-character relationships in videogames. *Game Studies: The International Journal for Computer Game Studies* (2002), vol. 2, nº 1 - <http://www.gamestudies.org/0102/newman/> (último acesso a 28 de Setembro 2016)
15. Eisler, H. & T. Adorno. *Composing for the Films*, Nova Iorque, Oxford University Press (1947)
16. Gorbman, C. *Unheard Melodies: Narrative Film Music*, Indiana University Press (1987)
17. Citron, M. J. Opera and Film. in Neumeyer, D. (ed.), *The Oxford Handbook of Film Music Studies*, Oxford University Press (2014), 44-71
18. Buhler, J. Ontological, Formal, and Critical Theories of Film Music and Sound. in Neumeyer, D. (ed.). *The Oxford Handbook of Film Music Studies*, Oxford University Press (2014), 188-225
19. Chion, M. *A Audiovisão*, Lisboa, Texto & Grafia (2011). M. Chion, no seu livro *A Audiovisão* (1990), discute os conceitos de *síncrise* e *valor acrescentado* relativamente ao tipo de relação estabelecido entre a imagem e o som e como esta é afectada (e afecta) a percepção do espectador, os quais podem também ser aplicados, até a uma certa extensão, nos videojogos.
20. Martin, P. The pastoral and the sublime in *Elder Scrolls IV: Oblivion*, *Game Studies: The International Journal for Computer Game Studies* (2011), vol. 3, nº 11. Relativamente à construção da sua narrativa e representação da dualidade entre “bem e mal”, P. Martin

- afirma que: “The treatment of good and evil is a staple of the modern fantasy genre, a genre that derives much of its stock of themes and imagery from the work of Tolkien. *The Lord of the Rings* (Tolkien, 1954 to 55) may be primarily figured as an anti-war novel, but war is primarily figured by Tolkien as an industrial intrusion on an idyllic phase.”
21. Adams, D. The Annotated Score – Part 1: The Fellowship of the Ring. *The Music of the Lord of the Rings*, New Line Productions Inc. (2005)
 22. Turino, T. Signs of Imagination, Identity, and Experience: A Peircian Semiotic Theory for Music. *Ethnomusicology* (1999), Vol. 43, No. 2
 23. Genette, G. *Palimpsestes: La littérature au second degré*, Paris, Seuil (1982)
 24. Um comentário do utilizador MARDUK no vídeo partilhado pelo canal JENS SCHLAU da música *Ministrel's Lament*, a faixa nº 5 da categoria “explore” da banda-sonora de *Oblivion*. O comentário e vídeo podem ser consultados aqui:
<https://www.youtube.com/watch?v=K-URrsISx-M>
 (último acesso a 28 de Setembro 2016)
 25. Comentário do compositor Jeremy Soule na música *Fall of the Hammer* da banda-sonora do jogo *Oblivion* publicado no *facebook* pessoal (<https://www.facebook.com/OfficialJeremySoule/posts/1017631854991338>) e referenciado na *wiki* da série *Elder Scrolls*, que pode ser acedido aqui:
http://www.uesp.net/wiki/Oblivion:Music#cite_note-JSFallHammer-7 (último acesso a 28 de Setembro 2016)
 26. Referido na *wiki* da série *Elder Scrolls*, que pode ser acedido aqui:
http://www.uesp.net/wiki/Skyrim:Music#cite_note-JSWindGuideYou1-8 (último acesso a 28 de Setembro 2016)
 27. Mais informações sobre esta linguagem, escrita e respectivo desenvolvimento aqui:
http://elderscrolls.wikia.com/wiki/Dragon_Language
 (último acesso a 28 de Setembro 2016)

Towards a Procedurally Generated Experience: A Structural Analysis of Quests

Antonio Machado
Universidade de Lisboa,
Instituto Superior Técnico,
INESC-ID
2780-990 Porto Salvo,
Portugal

antonio.machado@tecnico.ulisboa.pt

Pedro Santos
Universidade de Lisboa,
Instituto Superior Técnico,
INESC-ID
2780-990 Porto Salvo,
Portugal

pedro.santos@tecnico.ulisboa.pt

João Dias
Universidade de Lisboa,
Instituto Superior Técnico,
INESC-ID
2780-990 Porto Salvo,
Portugal

joao.dias@inesc-id.pt

ABSTRACT

In this paper, we present a quest structure, in the form of a grammar, obtained from an analysis of the main quests of a single player role playing game (RPG), namely *“The Witcher 3 - The Wild Hunt”*. This grammar extends a previously presented analysis by other authors on MMORPG quests. This extended grammar is suitable for application in single player RPGs. The grammar allows the procedural generation of quests in any RPG wanting more intricate quests. We believe that this extension makes the previous grammar more expressive. And it will bring us closer to being able to represent and procedurally generate quests that are equal to human authored ones.

Author Keywords

Procedural generation; interactive storytelling; RPG; quests; story; NPC; player.

INTRODUCTION

Computer Role Playing Games (CRPGs), commonly referred to as Role Playing Games (RPGs) are a video game genre where a player embodies a story world character (and/or several, commonly referred to as party) and has to overcome a series of linked challenges, ultimately achieving some overarching goal or the conclusion of a central storyline. Furthermore, the player is allowed to develop his/her character(s) through consequential decisions. RPG's are known for being content-heavy games, possessing vast worlds with various non-playable characters (NPCs), as well as intricate storylines and side-quests [10]. The process of this content creation takes a considerable amount of time and money [12]. It's consumption though is much faster, and after a player completes every main quest and side-quest, the game's replay value drops off considerably.

As a consequence, RPGs are perfect candidates for the application of Procedural Content Generation (PCG), which is

the use of computer algorithms for creating content that meets a set of evaluation criteria [10][16]. This becomes quite useful, when trying to produce content for the game industry, that is becoming more demanding [11]. According to Hartsook[10] there are two broad uses of PCG in games: content creation and adaptation of gameplay. By automatically generating content, it could offload the task of content creation, reducing the amount of work done by humans, making the development costs cheaper. Also, by learning players' information that can't be known during design-time, such as their preferences, desires and abilities, one could personalize the story and world of the game. And thus, maximizing pleasure and minimizing frustration and boredom[10].

PCG has become quite popular in recent years, currently being used in a variety of different subfields. Examples of generated content include: dungeons and maps (Binding of Isaac: Rebirth[13], Diablo series[6]); enemies (Left 4 Dead[17]), animation of character behaviour (Spore[1]), weapons (Borderlands 2[9]) and even whole universes (No Man's Sky[8]). Another area where PCG could be used is in generating interactive stories (examples given in the next section). A system based on interactive storytelling must be capable of generating interactive narratives in a coherent and believable fashion. This means that the sequence of events that constitute the overarching story must be causally and temporally coherent and characters that partake in these events must act in a believable manner[3][14]. In storytelling systems, characters are perceived as being believable when the actions they execute are motivated by their desires and intentions and these are consistent with the knowledge they possess about the current story world.

In the context of RPGs, the sequence of events that constitute the overarching story, can also be called quests. Quests are tasks given by NPCs to a player in the form of requests, that require the player to complete goals often in return for some reward. If quests are seen as the mere movement from one location to another in order to complete a goal, which involves character actions and dialogue, one can see that a sequence of this quests could shape a story. In games, story is the progression of the player through space[2][10]. The set of quests that are necessary to complete the game form up the main story. Additional side-quests are often offered to

the player to extend the gameplay[12]. Having a system that procedurally generates these quests can potentially increase the variability and replayability of games.

This paper will focus solely on quest structure, and presents the results obtained from our structural analysis of the main story quests from "The Witcher 3 - The Wild Hunt", an acclaimed single player RPG game. This analysis departs from and further extends the work done by Doran and Parberry [4] in their structural analysis of several MMORPG quests. We consider a quest's structure to be all actions performed by the player, since the moment the quest is given to him/her, until the quest's goal is achieved. Using this structure, we can procedurally generate a variety of intricate quests and apply them in a RPG game. The remainder of this paper is divided into four sections: 1. review of several approaches; 2. presentation of the results of the structural analysis of the main quests from "The Witcher 3 - The Wild Hunt" (Witcher); 3. an example of a quest from Witcher, based on the results obtained; 4. conclusions and future work.

RELATED WORK

To solve the problem of plotline adaptation, Li and Riedl[12] present an offline algorithm that, given a main plotline, consisting of a sequence of quests, a library of quests, and a set of player requirements, produces "a sound, coherent variation" of the original plotline. Preserving the human authors' intent, while meeting player requirements. The complete plotline is represented by a partially ordered, hierarchical plan composed of events that will unfold in a virtual world. These events can be within and outside quests, and are represented by actions performed by the player, non-player characters, or even by natural occurrences in the virtual world. Events possess preconditions, that must be satisfied, and effects, that become true. Causal relationship between two events is established through causal links via some condition that needs to be satisfied. They allow abstraction hierarchies, through decomposition of abstract events into less abstract ones.

In their approach quests are represented as "top-level" abstractions. Quests have only one effect, the acknowledgement of its completion, and it may or may not have preconditions. Quests are then decomposed into two abstract events: a task and a reward, which are further decomposed into basic actions. A quest that requires the player to hunt down a witch, can be decomposed in the following way: the player would first have to get a water bucket to pour on the witch, ultimately killing her (the task event), in order to acquire the trust of the king (the reward event). In between, events like the witch dropping her shoes, the player picking them up and showing them to the king as proof of the achievement, would complete the plotline. Narrative soundness and coherence, are then guaranteed, through the satisfaction of all preconditions of an event, and through the causal links connecting each event, thus creating a path that leads to a significant outcome.

The game plot adaptation algorithm takes the partial-order plan described, as well as the set of player preferences. The search is conducted by adding and removing events until success criteria are met. Once complete, the resulting story structure is converted and sent to GAME FORGE system[10] to

render a world that supports the story and executes the game. Indeed, although Li and Riedl[12] are capable of producing quests with a certain amount of control, based on players' requirements, they are still dependent on human authoring for the sequence of quests that constitute the plotline. Furthermore, the quests are customized and generated at the start of the game, as opposed to being generated while the player is in play.

Doran and Parberry [4][5] did a structural analysis of almost 3000 human-authored quests from several Massive Multi-player Online Role Playing Games (MMORPG). The analysis showed a common structure shared by human-authored quests, "changing only details such as settings, but preserving the relationship between actions". They observed structural patterns in quests, which occurred in predictable situations, each with its own implicit preconditions and effects.

They first observed that quests can be categorized into 9 distinct NPC motivations: Knowledge, Comfort, Reputation, Serenity, Protection, Conquest, Wealth, Ability and Equipment. They believe the use of motivations to be essential for ensuring intentionality in the generation and giving of quests. Quests are thus intended to represent a NPC's prime concern. Each of these motivations contains 2-7 motivation-specific strategies. In turn, each of these strategies is composed of a sequence of 1-6 actions, that the player must perform. Each action is further defined as either an atomic action performed by the player, or a recursive sequence of other actions or action variants[4]. The quest structure is represented in the form of a grammar, in which terminal symbols are atomic actions and non-terminal symbols are action rules, that extend to further actions or action rules. The sequence of actions, that the player is required to perform in order to complete the quest, can be viewed as the leaves in a tree, with the root representing the entire quest [4]. Actions can also be replaced by sub-quests, that use the same structure.

With this structure, made from the extracted rules and commonalities of the analysed quests, they are able to demonstrate a prototype system that procedurally generates quests, which in their view are appropriate for use in RPGs. The generator starts with an NPC motivation, from it the generator consults the list of specific strategies, selects one and creates a quest that addresses the motivation. The generator was written in Prolog, due to its "ability to backtrack and try alternative solutions"[4].

As a rough generalization, single player RPGs tend to focus more on the journey(story), which plays a central role in these type of games. At the end the player either explores the open world or simply restarts the game in a new save file. As a contrast, in MMORPGs this journey feels more like a grind that the player wants to complete as quickly as possible, in order to get to the endgame content, where the true game begins. Quests generated using the structure previously described, which was used by Doran and Parberry on their prototype generator [4], are solely based on MMORPGs.

So as stated before, according to Doran and Parberry's structural analysis, human authored quests have a shared structure.

Having this in mind, we tried to analyse quests from single player RPGs, which tend to have a strong focus on the story component of the game, using the rules defined by Doran and Parberry [4]. We chose “*The Witcher 3 - The Wild Hunt*”, since it was received with critical acclaim and was a financial success, having also won several awards for Game of the Year from multiple publications. For the purpose of this paper, only the main story quests were analysed, which are in a total of 58, since side-quests are more similar to MMORPG quests.

STRUCTURAL ANALYSIS

For the structural analysis, 58 main story quests from “*The Witcher 3 - The Wild Hunt*” (Witcher) were examined, in order to determine whether they also shared the structure extracted by Doran and Parberry from their own analysis. Quest descriptions were obtained from sites like “*The Witcher Official Wikia*” [20], VG24/7 [18], and from watched walkthroughs from several Youtube channels like TheRealCheatCC [15], GameRiot [7] and VGFAQ [19]. Quest descriptions consisted of the sequence of actions the player had to perform, NPC dialogue and causal and temporal relationships between the different quests.

Using the descriptions from “spoiler” sites and the resulting quest structure from the analysis described by Doran and Parberry [4], we undertook an attempt to represent Witcher quests. Alas, representing these main quests proved to be more difficult than expected. Indeed, the NPCs that gave out the quests, shared the same or similar motivations previously described, but since the action rules Doran and Parberry defined had some limitations, it was impossible to fully describe Witcher quests. In order to come up with the necessary changes to counter these limitations (both described in the upcoming paragraphs), each quest was analysed in the following way: first it was necessary to record every player action in the respective quest; second a tree was built for each quest, in the manner described in the related work section, using the rules defined by Doran and Parberry [4], while adding the changes required to represent each quest; third, the changes made were extracted and added to the new set of rules, the results of which can be seen in Tables 1, 2 and 3. Changes made are highlighted in bold. The sequence of actions and rules are written in Backus Normal Form, a notation technique for grammars, same as in [4].

When comparing with [4], the first change worthy of noting, regards the last action of a quest’s strategy, specifically “give” and “report” actions. During our analysis, we observed that some Witcher quests required the player to give or report something, as a last step before completing a quest. But some strategies defined in [4] don’t allow this. The opposite also happened, where strategies defined in [4], have a last action give/report, but in the game the player isn’t required to do any of those. To give a few examples: in the short main story quest “Disturbance”, the player has to explore the castle, to find and remove an object (“repair”), that is messing with one of Yennefer’s spells. After its removal the player is required to report back to Yennefer. This sequence isn’t fully represented in [4], because it is missing the report action. A second example would be the main story quest “The

Sunstone”, which requires the player to find and gather the lost item Sunstone. Using Doran and Parberry’s structure, the player would be obliged to give the Sunstone after it was gathered. However, this quest finishes as soon as the Sunstone is acquired. This was a recurrent problem, so to counter this obstacle, we decided to put in almost every strategy, one of two action rules, namely a <give> or a <report>. This way, it is now possible to decide during generation, whether a quest’s final action should require the player to either report a quest’s completion, give an item back, or neither.

The second change resides in the <goto> set of rules (see Table 2). According to Doran and Parberry’s rules, it wasn’t possible to be given a sub-quest without having to learn something (see Table 2, rule 9 and rules 12-14), which is something that happened often in The Witcher 3. So it was necessary to add a <prepare> rule, that offered this possibility (see Table 2, rules 10 and 16). Also the rule that required learning (rule 10 from Table 2) no longer has a mandatory “goto” action. Which limited the order in which certain events could occur. Now, the new <goto> action rule can be expanded to offer more possibilities to the quest. It was also added the action rule <rescue> (see Table 2, rules 33-36), in order to allow the player to simply free a character, and not having to escort it every single time. The action rule <defeat> (see Table 2, rules 27 and 28) was added, so it could be possible to decide during generation whether a strategy would require the player to either kill or merely damage an enemy. Before strategies had one of these actions imposed, which conflicted with the representation of some of the Witcher quests, that had one of those strategies. In some Witcher quests, where it was only required for the player to damage someone or something, the rules defined by Doran and Parberry would instead have the player kill someone or something, and vice-versa. An example of this is the quest “Bald Mountain”. After the death of their mentor Vesemir, Geralt (the player) and Ciri bloody for vengeance, track down Vesemir’s killer Imlerith and ultimately kill him. Using Doran and Parberry’s set, it wouldn’t be possible to represent this. In their set, the strategy “Revenge, Justice” from the “Serenity” motivation has a mandatory “damage” atomic action, while what we seek is a “kill” atomic action.

Finally, for the actions (see Table 3), we added four new atomic actions that can be performed by the player. During the analysis we encountered some actions that were not represented in their set of actions. The most significant ones being “examine” and “follow”. Doran and Parberry’s rules only considered learning either through listening to a character, after doing a sub-quest for said character, or by reading a book. After analysing Witcher quests it was clear that one can also learn by examining clues or objects. One example is the quest “Novigrad Dreaming”, where a ghost leaves drawings that, after being examined, show the player what he/she has to do next. The action “follow” also appears a lot, practically in every quest. While following a character, the player has

Motivation	Strategy	Sequence of Actions
Knowledge	Deliver item for study Spy Interview NPC Use item on field	<get> <give> <goto> spy <report> <goto> listen <report> <get> <goto> use <give>
Comfort	Obtain luxuries Kill pests	<get> <give> <goto> <defeat> <report>
Reputation	Obtain rare items Kill enemies Visit dangerous place	<get> <give> <goto> <defeat> <report> <goto> <report>
Serenity	Revenge, Justice Capture Criminal Check on NPC (1) Check on NPC (2) Recover lost/stolen item Rescue NPC	<goto> <defeat> <report> <goto> <capture> <report> <goto> listen <report> <goto> take <give> <get> <give> <goto> <rescue> <report>
Protection	Attack threatening entities Capture Criminal Treat or Repair (1) Treat or Repair (2) Create Diversion (1) Create Diversion (2) Assemble fortification Guard entity Recruit	<goto> <defeat> <report> <goto> <capture> <report> <get> <goto> use <report> <goto> repair <report> <get> <goto> use <report> <goto> damage <report> <goto> repair <report> <goto> defend <report> <goto> listen <report>
Conquest	Attack enemy Steal stuff Recruit	<goto> <defeat> <report> <goto> <steal> <give> <goto> listen <report>
Wealth	Gather raw materials Steal valuables for resale Make valuables for resale	<goto> <get> <report> <goto> <steal> <give> <goto> repair <give>
Ability	Assemble tool for new skill Obtain training materials Use existing tools Practice combat Practice skill Research skill (1) Research skill (2)	<goto> repair use <get> use <goto> use <goto> damage <goto> use <get> use <report> <get> experiment <report>
Equipment	Assemble Deliver supplies Steal supplies Trade for supplies	<goto> repair <give> <get> <give> <steal> <give> <goto> exchange

Table 1. Strategies for each NPC's motivation.

#	Rules	Explanation
0.	<Quest> ::= <Knowledge> <Comfort> <Reputation> <Serenity> <Protection> <Conquest> <Wealth> <Ability> <Equipment>	This is the root of a quest, which expands into one of the 9 motivations. Which will eventually be expanded into one of the strategies, specific to said motivation.
1.	<subquest> ::= <goto>	Go someplace.
2.	<subquest> ::= <goto> <QUEST> <goto>	Go perform a quest and return.
3.	<goto> ::= ε	You are already there.
4.	<goto> ::= goto	Go to a known location.
5.	<goto> ::= wait	Wait at a location for someone or something.
6.	<goto> ::= explore	Just wander around and look.
7.	<goto> ::= follow	Follow somebody or something.
8.	<goto> ::= stealth	Sneak by someone.
9.	<goto> ::= <learn> <goto>	Find out where to go and go there.
10.	<goto> ::= <prepare> <goto>	Prepare before going somewhere.
11.	<learn> ::= ε	You already know it.
12.	<learn> ::= <goto> <subquest> listen	Go someplace, perform a subquest, get info from NPC.
13.	<learn> ::= <goto> <get> read	Go someplace, get something and read what is written in it.
14.	<learn> ::= <get> <subquest> <give> listen	Get something, perform a subquest, give to NPC in return for info.
15.	<learn> ::= <goto> <subquest> examine	Go someplace, perform a subquest, examine something.
16.	<prepare> ::= <goto> <subquest>	Go someplace and perform a subquest.
17.	<get> ::= ε	You already have it.
18.	<get> ::= <steal>	Steal it from somebody.
19.	<get> ::= <goto> gather	Go someplace and pick something up that's lying around.
20.	<get> ::= <goto> take	Go someplace and take something.
21.	<get> ::= <goto> <get> <subquest> <goto> exchange	Go someplace, get something, perform a subquest for somebody, return and exchange.
22.	<steal> ::= <goto> stealth take	Go someplace, sneak up on somebody and take something.
23.	<steal> ::= <goto> <defeat> take	Go someplace, defeat somebody and take something.
24.	<capture> ::= <goto> use capture	Go someplace, use something to capture somebody.
25.	<capture> ::= <goto> damage capture	Go someplace, damage to capture somebody.
26.	<capture> ::= <goto> capture	Go someplace and capture somebody.
27.	<defeat> ::= <goto> damage	Go someplace and damage somebody.
28.	<defeat> ::= <goto> kill	Go someplace and kill somebody.
29.	<report> ::= ε	There is nothing to report.
30.	<report> ::= <goto> report	Go someplace and report to somebody.
31.	<give> ::= ε	There is nothing to give.
32.	<give> ::= <goto> give	Go someplace and give something to somebody.
33.	<rescue> ::= free	Free somebody from imprisonment.
34.	<rescue> ::= <defeat> free	Defeat somebody and free somebody from imprisonment.
35.	<rescue> ::= escort	Escort somebody to someplace.
36.	<rescue> ::= <defeat> escort	Defeat somebody and escort a different somebody to someplace.

Table 2. Action rules in BNF.

#	Action	Pre-condition	Post-condition
1.	ϵ	None.	None.
2.	capture	Somebody is there.	They are your prisoner.
3.	damage	Somebody or something is there.	It is more damaged.
4.	defend	Somebody or something is there.	Attempts to damage it have failed.
5.	escort	Somebody is there.	They will now accompany you.
6.	examine	Somebody or something is there.	You have information about it.
7.	exchange	Somebody is there, they and you have something.	You have theirs, they have yours.
8.	experiment	Something is there.	Perhaps you have learned what it is for.
9.	explore	None.	Wander around at random.
10.	follow	Somebody or something is there.	You will now accompany them.
11.	free	Somebody is there.	They are no longer prisoner.
12.	gather	Something is there.	You have it.
13.	give	Somebody is there, you have something.	They have it, you don't.
14.	goto	You know where to go and how to get there.	You are there.
15.	kill	Somebody is there.	They are dead.
16.	listen	Somebody is there.	You have some of their information.
17.	read	Something is there.	You have information from it.
18.	repair	Something is there.	It is fixed, built or resolved.
19.	report	Somebody is there.	They have information you have.
20.	spy	Somebody or something is there.	You have information from it.
21.	stealth	Somebody is there.	Sneak up on them.
22.	take	Somebody is there, they have something.	You have it, they don't.
23.	use	Somebody or something is there.	It has affected characters or environment.
24.	wait	None.	Wait for something to happen.

Table 3. Atomic actions.

the opportunity to learn a bit of the character's backstory and possible relationship with the main character Geralt. Actions "wait" and "free" were also added, the first rarely appearing, while the second can be seen as an alternative to simply escorting a character, after being rescued. This action appeared more often than the "escort" action in the Witcher's main story quests. A good example for the use of the action "free", is the quest "A Poet Under Pressure". Here the player has to rescue the halfling Dandelion. After following a Witch Hunter that fled from a failed ambush, the player enters the house where Dandelion is being held captive. After you defeat the Witch Hunter, the player is able to free Dandelion and report to Irina. This quest is also a good example for the addition of the action rule <defeat>. In Doran and Parberry's strategies, the "Rescue NPC" strategy (see Table 1, motivation Serenity) required the player to "damage" the captor, whilst in this quest the player is required to "kill" the captor.

QUEST EXAMPLE

In this section we analyse the quest "The Beast of White Orchard", one of "The Witcher 3" main story quests, using the new set of rules (see Figure 6 for full quest). The quest is given out by a "Nilfgaardian" Commander with the promise of information about the witch Yennefer, which Geralt (the player) is currently looking for. The Nilfgaardian army is having trouble with a Griffin, that has been randomly attacking its soldiers, and the Commander wants Geralt to kill it. Looks simple, but this quest requires the player several preparatory steps. First the player has to find information by talking to a hunter about the griffin. Information like where its current location is or why it is on a rampage. He/she also has to gather

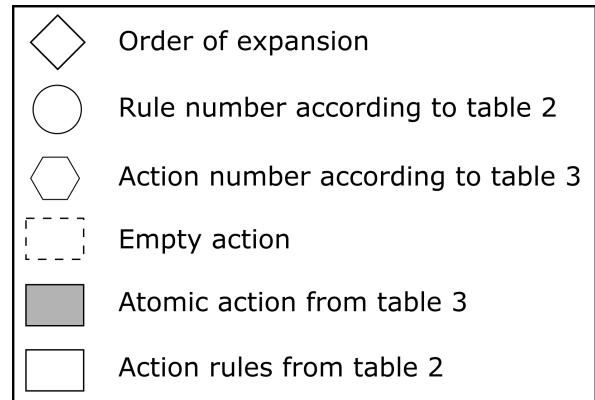


Figure 1. Key to all other Figures.

a plant with a strong scent that attracts it, by talking with a herbalist. In a way the player needs to prepare before the encounter with the griffin. This quest can be seen as having the motivation "Comfort", using the strategy "Kill Pests", which starts with the sequence of actions "<goto> <defeat> <report>" (see Figure 2, and Table 1).

The first action rule <goto>, implies that the player must go to the griffin's location. In Doran and Parberry's rules, the player would either have to learn the location, in case it was unknown, or explore. In this quest, instead, it was required to make preparations before facing the griffin, namely to learn about the griffin and gather something to lure it. So in this case we use rule number 10 (from Table 2) "<prepare>

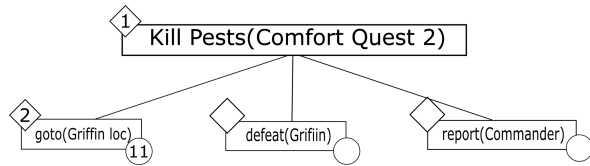


Figure 2. Initial strategy of example quest.

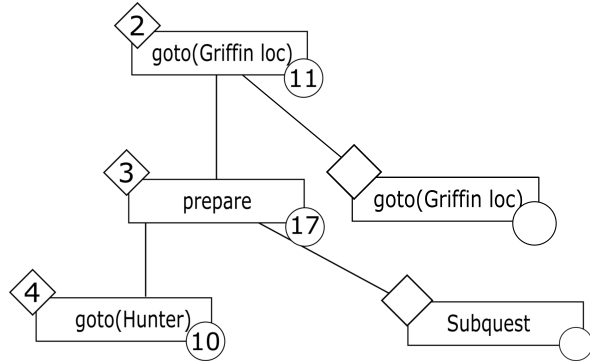


Figure 3. Expansions of <goto> (rule number 11) and <prepare> (rule number 17), both newly added.

<goto>". The action <prepare> will be expanded to "<goto> <subquest>" (rule number 16)(see Figure 3).

The next <goto> (4th expanded action), requires the player to go to the site where Nilfgaardian soldiers were attacked, but first he/she has to talk to the hunter, to guide the player there. So the <goto> is expanded to "<learn> <goto>" using rule number 0. The <learn> is then expanded using rule number 12 "<goto> <subquest> listen". The player must first go to the Hunter's house to find that he isn't there, which requires the player to explore and examine clues that direct the player to the Hunter's location (see Figure 4). Here is the first instance where the use of Doran and Parberry's rules, is unable to represent the quest. It wouldn't be possible for the player to find the hunter, without the atomic action "examine" and the newly added rule number 15, which isn't represented in their set of rules.

After finally reaching the Hunter (12th expansion), the <QUEST> action rule is then expanded to Comfort motivation strategy "Kill Pests". Here the hunter asks the player to kill some wild dogs that are troubling him (13th-20th expanded actions). After reporting to the hunter the player must follow him to the site where the soldiers were attacked. Again this atomic action isn't present in the rules defined in [4]. Once at the site, the player has again to examine some clues, that lead to tracks that must be followed. Finally leading to the griffin's nest, where the player learns the final details about the griffin (21st-29th expanded actions)(see Figure 5). Notice that if we were using the set of rules defined by Doran and Parberry, these sequence of actions wouldn't be possible. Instead we would have an atomic action "goto", that would make the player go directly to the nest, alternatively to finding his/her way over there.

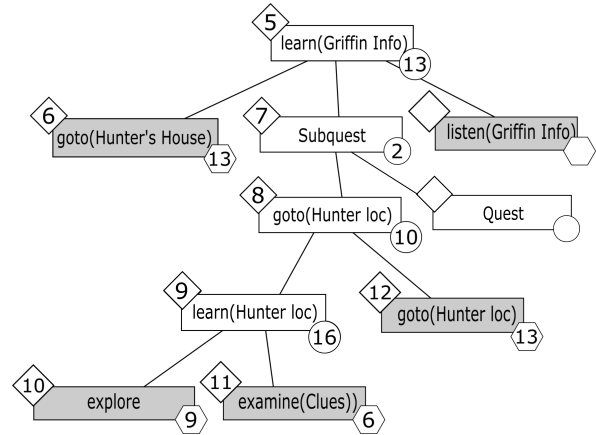


Figure 4. Expansions 5 to 12, use of newly added action "examine".

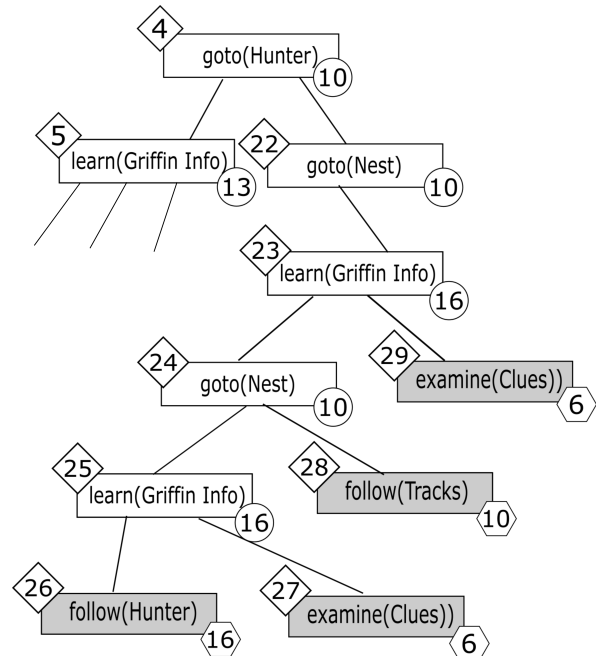


Figure 5. Expansions 22 to 29. Note expansion of <goto> after <learn> is closed and the use of newly added action "follow".

Thus closes the <goto> action rule expanded after <prepare> (3rd expanded action)(see Figure 3). Now the <subquest> is expanded using rule number 2 (from Table 2), "<goto> <QUEST>". Before facing the griffin, the player must still gather buckhorn to be used as lure. But first he/she must learn its location. The player must first talk to a herbalist to get this information. This is all represented through expanded actions 32-36 (see Figure 3). Although it wasn't given by any specific NPC, the gathering of the buckhorn is represented through the expansion of <QUEST> using the strategy "Gather Raw Materials" from motivation Wealth, which can be resumed to a simple atomic action "gather", since the player is already at the location (see Table 1).

Having gathered every bit of information and the necessary lure, it is now time to move to the griffin's location and ultimately defeat it. The <goto> from the original strategy ends with an atomic "goto" (see Figure 3), continuing with the expansion of the action rule <defeat>(see Figure 2). As stated before, the action rule <defeat> was added, so it would be possible for a generator to decide whether a strategy would have the player kill or damage an enemy. In the end, both actions can be summarized as defeating an opponent. The choice of which action to perform, could then be given either to the player or to the NPC giving out the quest. In this case, the <defeat> rule is expanded using rule 28 "<goto> kill" (see Table 2).

The next step for the player, is to use the previously gathered buckhorn to lure the griffin out. Since it isn't required to learn anything, we expand the <goto> action using rule 10, <prepare> <goto>. The <prepare> action rule is then expanded using rule 16. Now we have "<goto> <QUEST> <goto>", both <goto> are empty since the player is already where he/she needs to be, and <QUEST> is expanded using the strategy "Use existing tools" from Ability motivation, to an atomic action "use"(see Table 1 and Figure 6). Since the player isn't required to move, we are left only with the decisive action of killing the griffin (51st expanded action). Finishing with the <report> action rule, which is expanded using rule 30, "<goto> report"(see Table 2). The player most now return to the Nilfgaardian camp and report to the Commander (52nd-54th expanded action).

As can be observed, in figure 6, all required actions were successfully represented using this new set of rules. The addition of 4 new atomic actions, as well as the addition of action rules <report> and <give> to the strategies, was simply to help represent certain sequences of actions executed by the player in "The Witcher 3". The biggest difference, and probably the most influential, was the removal of the atomic actions "goto", after the <learn> and <QUEST> actions rules (see rules 2 and 9 from Table 2). These atomic "goto"s were restricting greatly the order in which actions could be performed. Their removal, allows for more variability, but we lose control over the size of the generated quest. Nonetheless, it is possible to restrict the expansion of these action rules. By means of limiting the depth of the tree, or simply by giving a probability

for expanding a rule, making some rules less or more likely in certain depths.

CONCLUSION

In this paper we introduce adjustments to the quest structure, defined by Doran and Parberry[4], in their analysis of MMORPG quests. These adjustments were based on our study of the 58 main quests from the prize-winning single player RPG game "*The Witcher 3 - The Wild Hunt*". In their paper Doran and Parberry concluded that further work was necessary to prove the capabilities of their generator in producing quests equal in quality as human-authored ones. We believe that our adjustments to the original structure, make it more expressive. Since no tests were conducted with human players, it is impossible to determine the quality of quests based on this structure. Nonetheless, we hope that this structure gives a significant contribution in making procedurally generated quests closer to the ones written by humans. Note, however, that through our analysis we found that Witcher quests aren't completely similar to the MMORPG quests. Thus, we suggest the study of other single player RPGs to further improve this quest structure, hoping it will help close the remaining gap.

The quest structure defined in this paper is to be implemented in a game experience, currently in development. Our goal is to create a system, that uses a procedural approach to story generation. The system will simulate a story world, where a structured narrative can emerge through the interactions between NPCs and the player in the form of quests, based on the structure presented here. NPCs will be responsible for reasoning about which actions are more appropriate for achieving a goal, select them and create a quest that is motivated by their own intentions. Thus, giving a greater sense of realism. Additionally, the system will also use a learned player model to adapt the subquests to the player's preferences. We hope this system will be able to improve both the chances of re-playability and player enjoyment, by offering the player more customized content variety.

ACKNOWLEDGEMENTS

This work was supported by national funds through Fundação para a Ciência e a Tecnologia (FCT) with reference UID/CEC/50021/2013.

REFERENCES

1. Electronic Arts. 2008. Spore. (2008).
2. Calvin Ashmore and Michael Nitsche. 2007. The Quest in a Generated World. (2007), 503–509.
3. Michael Brenner. 2010. Creating Dynamic Story Plots with Continual Multiagent Planning. *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence (AAAI-10)* Lochbaum 1998 (2010), 1517–1522. <http://www.aaai.org/ocs/index.php/AAAI/AAAI10/paper/download/1709/2245>
4. Jonathon Doran and Ian Parberry. 2011a. A prototype quest generator based on a structural analysis of quests from four MMORPGs. *Proceedings of the 2nd International Workshop on Procedural Content*

- Generation in Games - PCGames '11* (2011), 1–8. DOI: <http://dx.doi.org/10.1145/2000919.2000920>
5. J Doran and I Parberry. 2011b. Towards procedural quest generation: A structural analysis of {RPG} quests. *ProcDoran, J., & Parberry, I. (2011). Towards procedural quest generation: A structural analysis of {RPG} quests. Proceedings of the 2nd International Workshop on Procedural Content Generation in Games. eedings of the 2nd International Workshop on Procedur* (2011).
 6. Blizzard Entertainment. 1996-2014. Diablo Series. (1996-2014).
 7. GameRiot. 2015. (2015). <https://www.youtube.com/user/GameRiotArmy>
 8. Hello Games. 2016. No Man's Sky. (2016).
 9. 2K Games. 2012. Borderlands 2. (2012).
 10. Ken Hartsook, Alexander Zook, Sauvik Das, and Mark O. Riedl. 2011. Toward supporting stories with procedurally generated game worlds. *2011 IEEE Conference on Computational Intelligence and Games, CIG 2011* (2011), 297–304. DOI: <http://dx.doi.org/10.1109/CIG.2011.6032020>
 11. Mark Hendrikx, Sebastiaan Meijer, Joeri V A N D E R Velden, and Alexandru Iosup. 2011. Procedural Content Generation for Games : A Survey. 1, February (2011), 1–24. DOI: <http://dx.doi.org/10.1145/0000000.0000000>
 12. Boyang Li and Mo Riedl. 2010. An Offline Planning Approach to Game Plotline Adaptation. *Association for the Advancement of Artificial Intelligence-Aiide* (2010), 45–50. <http://www.aaai.org/ocs/index.php/AIIDE/AIIDE10/paper/viewFile/2125/2542>
 13. Nicalis. 2014. Binding of Isaac: Rebirth. (2014).
 14. Mark O. Riedl and R. Michael Young. 2010. Narrative planning: Balancing plot and character. *Journal of Artificial Intelligence Research* 39 (2010), 217–268. DOI: <http://dx.doi.org/10.1613/jair.2989>
 15. TheRealCheatCC. 2015. (2015). <https://www.youtube.com/user/TheRealCheatCC/search?query=witcher>
 16. Julian Togelius, Georgios N Yannakakis, Kenneth O Stanley, and Cameron Browne. 2011. Search-based Procedural Content Generation : A Taxonomy and Survey. (2011), 1–15.
 17. Valve. 2008. Left 4 Dead. (2008).
 18. VG24/7. 2016. The Witcher 3: Wild Hunt guide and walkthrough. (2016). <http://www.vg247.com/2016/05/30/the-witcher-3-guide-walkthrough-pc-ps4-xbox-one-wild-hunt/>
 19. VGFAQ. 2015. (2015). <https://www.youtube.com/user/VGFAQ>
 20. The Witcher Official Wikia. 2015. The Witcher 3: Wild Hunt - Guide to Main Quests. (2015). http://witcher.wikia.com/wiki/The_Witcher_3:_Wild_Hunt_-_Guide_to_Main_Quests

A representação subjetiva dos signos culinário e a polícia no jogo Max Payne 3. Uma análise semiótica da representação imagética em jogos de video game

Carlos William F. de Lima
Universidade Anhembi Morumbi
São Paulo - ESPM, Brasil
Carlos.william.sp@gmail.com

RESUMO

O presente artigo, apresenta as relações socioculturais da cidade de São Paulo, com representações imagéticas no jogo Max Payne 3 (2012). Durante o jogo, são encontrados objetos usados pela polícia sobre as mesas, na tentativa de ambientar melhor a delegacia, tal qual seria na cidade, buscando reproduzir quaisquer elementos que fazem parte da vida cotidiana dos policiais, porém, ao mesmo tempo, encontramos um prato típico da cidade, a coxinha de frango. Essa representação levou-nos a buscar entendimento de como e por que um prato que se tornou um dos símbolos da cidade, pode ser atrelado como representação semiótica da polícia militar de São Paulo. Para isso, traçamos uma abordagem histórica da coxinha e o alinhamento com a história do jogo e encontramos a análise de Leonardo Lima, que estudou a produção de subjetividade nos jogos digitais, e a semiótica Perciana, para melhor compreendermos o uso desse símbolo durante o jogo.

PALAVRAS CHAVES

Games, Culinária, Coxinha de frango paulista, Polícia Militar, Max Payne3

INTRODUÇÃO

Assim como a culinária, os jogos digitais começam a fazer parte da cultura de muitos povos. No caso especificamente do Brasil, temos uma cultura rica e amplamente missegenada, pois, como temos registros históricos, a influência das mais diversas culturas que aqui portaram trouxeram consigo inúmeros exemplos de sua culinária e entre elas, conhecemos a coxinha de frango.

Neste artigo, abordamos a influência dessa iguaria, até mesmo na cultura digital e especificamente no jogo de videogame Max Payne3, que, além de aparecer no jogo como um elemento de cena, também tem sua representatividade apresentada, como um elemento de ligação semiótico, entre a polícia e o jogador. Este artigo,

pretende apresentar, dados históricos e fazer uma análise semiótica sobre o signo coxinha.

Além de traçarmos uma linha histórica sobre a coxinha, também trabalhamos com o jogo Max Payne 3, que conseguiu traduzir em forma de imagem, as características inerentes dessa representação da culinária de São Paulo.

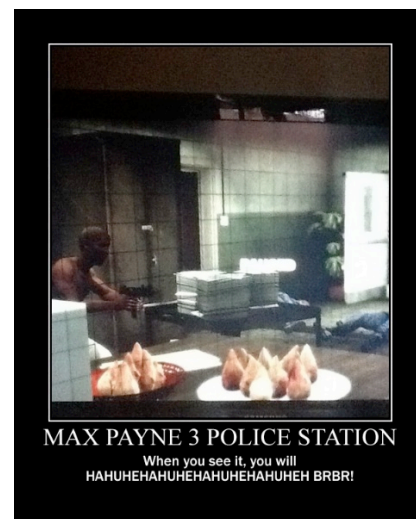


Figura 1. Coxinha de frango na delegacia de polícia.

A COXINHA DE FRANGO

Não se sabe ao certo a origem da coxinha de frango na culinária paulistana, mas, em busca de informações mais concretas, encontramos duas histórias mais próximas da realidade de sua criação.

A primeira, no livro de Nadir CAVAZIN [1], Histórias e Receitas, conta uma história, que pode nos dar um caminho para entendimento do quão importante foi a coxinha de frango, um salgadinho largamente encontrado na cidade de São Paulo[12][13].

A história retrata um período já no fim do império brasileiro que, na cidade de Limeira, o imperador Dom Pedro II e da

imperatriz Tereza Cristina de Limeira. Foram feitas visitas formais teriam acontecido em 1876 e em 1886, para participação da missa na recepção à ilustres cidadãos na Fazenda Morro Azul. [1]

Outra história que também ilustra bem o surgimento da coxinha de frango pode ser encontrada [2], em que ele descreve a criação da coxinha no século XIX e como um prato tipicamente paulistano, mesmo embora seja possível encontrá-lo na culinária portuguesa também.

Sua história está diretamente ligada à fase industrial paulistana, quando, nas portas das fábricas, eram vendidas coxas de frango in natura, assadas e prontas para o consumo da classe trabalhadora a qualquer tempo. Como sua produção era baixa e o tempo de deterioração muito rápido, buscou-se uma alternativa para que pudessem ser vendidas com mais segurança e tempo maior de validade. A escolha pela farinha de trigo deu-se em função de ser um ingrediente largamente usado na culinária paulistana, já que fora introduzida pela culinária portuguesa e italiana em seus pratos, e dessa forma, produzindo as mesmas coxas de frango em um novo formato, seria possível atender um maior número de pessoas e com menos frango em sua composição.



Figura 2 - Coxinha de frango, com seu formato característico e tamanho proporcional, gerando um padrão visual repetitivo.

Hoje, esse salgado é encontrado em quase todo território brasileiro.

A POLÍCIA DA CIDADE DE SÃO PAULO

O conceito de polícia tem sua origem na Grécia. Em [3], baseado na etimologia da palavra, encontramos a seguinte definição, considerando que o papel da polícia tem como ponto mais importante para sua atuação, manter a cidade unida para garantir governabilidade.

Um aspecto que pode ser levantado é a distinção entre a polícia que rege a ordem e a que se faz seguir, pois há, dentro da organização do estado político, parte das autoridades policiais que determinam regras e a segunda parte que está em meio ao povo, que deve garantir o respeito a essas regras.

Podemos perceber, também segundo a análise de [3], que há outra determinação do termo polícia, emprestado pelos romanos e que também explica a atuação da polícia e suas atividades na sociedade. A essência da função

governamental consiste em definir fronteiras entre o público e o privado, por meio da produção de normas cujo o respeito é assegurado por órgão administrativos específicos, que utilizam, se necessário, o constrangimento físico.

Já no século XVIII, na Dinamarca, o conceito de polícia toma outro sentido, que se generalizará na Europa, em que polícia não mais remete ao que o cidadão atual chamaria de administração governamental, isto é, a regulamentação relativa à manutenção da ordem, da tranquilidade pública, da higiene, do comércio, do trabalho e outras divisões ligadas à administração pública, mas sim garantir a liberdade do cidadão, sem entraves para seu papel em sociedade.

Outro ponto a ser levantado é o uso da polícia não apenas como manutenção da ordem, mas também como força de obtenção e manutenção de poder [4] trata desse poder de forma clara em seu livro Vigiar e Punir, onde aborda a função repressora do estado por meio do uso de forças policiais.

No Brasil, o conceito de polícia é tido como a primeira força de natureza constitucional e que se destina a garantir e assegurar a proteção dos direitos legais e constitucionais dos cidadãos. A origem da polícia ocorre no Brasil Colônia, quando foram criadas as “companhias de ordenanças”, em 1570, e delas, muito tempo depois, em 1709, as “tropas pagas”, que tinham em seus quadros profissionais remunerados pelos seus serviços. Entre suas atividades, eram incumbidos de patrulhamento, rondas, condução de presos, combate à desordem e estavam subordinados aos governadores de província.

Logo, o papel da polícia, em diversas sociedades e períodos históricos, tem atuação direta na sociedade, porém, nos últimos anos, seu papel tende a ser apenas de organizar e zelar pela segurança e os direitos dos cidadãos, mas a hierarquia militar passa a ser mais forte. Em 1934, no Brasil, sua estrutura deixa de ser denominada de Brigada Policial para ser denominada Polícia Militar. Outro marco importante para a atuação da polícia militar surge durante a ditadura militar, entre 1964 e 1985, [15] momento em que as atividades e a competência das ações policiais foram ampliadas, passando de mantenedora da ordem pública e privada a força repressora junto a sociedade, inibindo qualquer forma de representação social e política [3]

Constatamos que o processo de treinamento do policial, também está presente nas modernas forças policiais e, no estado de São Paulo, essa força, ainda denominada por Polícia Militar do Estado de São Paulo, carrega o treinamento militar e a estrutura adequada a esse tipo de organização, como explica [3]

A polícia militar hoje é uma organização fardada, organizada militarmente, subordinada ao Governo do Estado, por meio da Secretaria de Segurança Pública.

A REPRESENTAÇÃO DOS SIGNOS NOS JOGOS DIGITAIS

Quando analisamos os jogos de videogame em um outro momento [5], norteados pelas análises da semiótica pierceana, podemos construir um entendimento do papel dos jogos digitais quanto ao agenciamento proporcionado pelo uso das imagens hiper-realistas e concluímos que o uso dessas imagens não era necessariamente tão importantes, pois, em alguns casos, para existir o agenciamento por parte do jogador, ela tem de existir pelo jogo, sua mecânica e como esse jogo faz relação direta com o jogador. Durante a atuação no jogo, o jogador se deparará com vários elementos semióticos e esses, terão maior ou menor impacto em sua leitura visual e por esse motivo, dado a interação semiótica, o jogador terá maior ou menor compreensão em relação ao signo apresentado.

Partindo disso, é possível compreender que as imagens apresentadas nos jogos também carregam um caráter cultural e, neste caso, fixamo-nos em *Max Payne 3 de 2012*, para referenciarmos o uso da imagem do salgadinho coxinha de frango, apontando para uma gíria popular atrelado à imagem da coxinha.

Em nosso entendimento, o foco desse estudo, debruçou-se sobre a coxinha de frango, pois ela tem sido atrelada à imagem da polícia militar do estado de São Paulo, muito provavelmente, desde os anos de 1980 e essa referência cultural é retratada no jogo de *Max Payne 3*, não de forma explícita, mas apenas percebida por quem conhece o significado, ou seja, o jogador que vive na cidade de São Paulo e que, em algum momento, teve contato com esse significado e consegue perceber essa subjetividade no jogo.

É possível encontrarmos uma analogia semelhante na cultura norte-americana, pois há uma referência aos policiais americanos à imagem dos Donuts¹, ou rosquinhas doces, encontradas no mercado americano e tão largamente representada nos filmes americanos, no entanto, não aprofundaremos essa analogia neste artigo, trataremos apenas do salgadinho coxinha. Para compreender as relações criadas com o salgadinho brasileiro em referência à Polícia Militar, necessitamos conhecer não mais a história do salgadinho apenas, já tratado no início deste artigo, mas nos indícios históricos de sua associação à imagem do policial militar, o representante mais próximo da corporação junto a sociedade.

LIMA [6] explica por que fazemos certas ligações com objetos e como a Semiótica pode nos ajudar nessa relação subjetiva dada ao objeto – especialmente aqui, essa associação da imagem da coxinha ao policial: O signo transpassa o ambiente e se recoloca em outra posição e

forma durante o jogo, mas seu significado é feito apenas pelo interpretante, no nosso caso, o jogador. Essa interpretação é dada por meio de diversas vivências e ela é que determina o entendimento desse signo, a coxinha de frango no jogo e seu significado.

O motivo da relação da imagem da coxinha ao policial militar pode ser explicado pelo fato de que a coxinha tem um formato regular e aplicado a todas unidades feitas, ou seja, elas são elementos iguais, diferentemente do que encontramos em outros pratos, quando na sua elaboração. A própria formulação desse prato torna-o diferente, mas, com seu formato em desenho de gota, parecido com a coxinha de frango, torna-o um elemento repetitivo e em muito se assemelha a um produto fabricado mecanicamente.

Talvez, pelo formato similar, tenha sido feita uma associação à corporação, pois os policiais usam as uniformes iguais, tornando-os únicos, assim como um prato repleto de coxinhas de frango. O que buscamos reforçar, aqui, é apenas a associação imagética, que não reflete a opinião deste autor.

Outra informação que encontramos foi a de que, nos anos de 1980, era muito comum os policiais serem vistos comendo coxinhas de frango em bares na cidade de São Paulo e isso também gerou a associação aos baixos salários da corporação à época, o que apenas possibilitava a compra desse salgado e, talvez por esse motivo, esse fato criou a “gíria” que ligou a imagem da coxinha de frango aos policiais militares e, a partir daí, o termo ficou tão subjetivamente ligado ao policial que gerou uma ligação imagética da polícia com a coxinha de frango.

Esse agenciamento da imagem da coxinha de frango ao policial, pode ser melhor explicado por [6], que nos dá informações valiosas sobre o agenciamento e a subjetividade, implicando em funcionamento de uma natureza estrapessoal e extra individual na representação de imagem de valor.

Baseados nessa afirmação, passamos a compreender melhor o porquê da associação da imagem da coxinha com a polícia militar, a partir de agora apresentamos o uso dessa imagem dentro do jogo e seus significados.

Isso nos leva a crer que a orientação para a compreensão da inserção do prato de coxinha dentro do jogo de videogame de *Max Payne 3* e o jogador que vive na cidade, e tem essa compreensão da gíria, pode compreender mais rapidamente o papel do elemento, coxinha de frango, aplicado ao jogo, assim que *Max Payne* envereda pelo cenário da delegacia policial.

¹ DONUT, DOUGHNUT, DÓNUTE OU ROSQUINHA É um pequeno bolo em forma de rosca (mais precisamente de toro), popular nos EUA e de origem incerta.



Figura 3. Max Payne entra na delegacia policial da cidade fictícia de São Paulo. O detalhe da imagem é o prato de coxinha sobre a mesa.

Certamente, as informações que surgiram em nossa análise dão como apontamento os resultados obtidos e apresentados neste texto, mas encontramos um desdobramento para explicar o significado subjetivo da associação da coxinha não só à polícia militar, mas também a outros segmentos da sociedade, como a classe social, que podemos perceber em meio à cultura paulista, porém decidimos não nos ater a esse aspecto, pois requereria uma análise ainda mais aprofundada e específica, além disso, o objetivo deste trabalho era compreender a ligação da coxinha com a polícia e sua representação dentro do jogo de *Max Payne 3*.

O contato do jogador com os jogos de videogame também está impregnado de suas relações pessoais, atravessado pelos convívios sociais, políticos, variáveis econômicas, sejam elas internas ou externas [6], criando em si as relações necessárias para a interpretação de objetos, apresentados a ele subjetivamente e ligados à cultura local.

Podemos perceber, na apresentação desses objetos, essa ligação, pois a produtora, quando decidiu fazer o jogo ambientando São Paulo, teve a preocupação de levantar dados culturais para apresentar esses elementos no jogo. O dado mais relevante é que esse signo só poderá ser reconhecido pelo público que conhece seu significado, portanto, o público que usa do jogo para se desconstrair, como é o caso do jogador americano, provavelmente não fará a mesma leitura do significado da coxinha sobre a mesa, pois, como sabemos, nossa cultura não é largamente

CONCLUSÃO

Todo o processo de construção imagética é fornecido pela própria produtora, durante o desenvolvimento do jogo. As imagens digitais tem como característica mais importante, a representação da realidade. A arte, a fotografia e até mesmo a literatura também cumprem esse papel, porém, os jogos digitais dão ao jogador, a Liberdade de exploração e vivência de diversas realidades, que não as suas.

O jogo *Max Payne 3*, traz em sua construção, uma cidade de São Paulo que não necessariamente é a que vemos, mas uma releitura de locais e principalmente de relações entre os personagens, o jogador e a cultura.

disseminada pelos meios de comunicação, assim como é feito com a cultura americana, propagada pela TV e pelo cinema.

Isso é reforçado por [7], que nos lembra que a mídia nos torna consumidores de informação, recortada, embalada e entregue de forma que possamos compreendê-la e apropriar-nos dela em nossa cultura. “No mundo da convergência das mídias, toda história importante é contada, toda marca é vendida e todo consumidor é cortejado por múltiplas plataformas de mídia”. (JENKINS, 2006)

Esse cortejo a que se refere [8], mesmo que nos apropriemos desses ditos, produtos, também vendemos alguns em troca e esse escambo cultural, de certa forma, reconstrói nossa subjetividade, dialogando com o interno e o externo, coexistindo com a cultura, a política, a sociedade e implicando o acoplamento do jogador. “O que queremos reter desses relatos é que o universo do jogo coexiste a uma realidade social, política, cultural e etc., as quais implicam-se os acoplamentos jogo-jogador” [6], na compreensão desses elementos no jogo e a direta interação com eles no mundo real.

O que podemos concluir, até aqui, é que o jogador tem papel fundamental na compreensão dos signos durante o jogo e a completa vivência cultural faz toda a diferença na compreensão de gírias, elementos e signos representativos de certas culturas e ações e, nesse jogo, os elementos apresentados foram cuidadosamente estudados para que o jogador brasileiro pudesse ter um agenciamento mais significativo em cada momento do jogo.

Antonio [9] também demonstra como esse processo é atrelado ao capitalismo, no sentido de mostrar como a produção está em meio às diversas mídias, mas que, de certa forma, objetiva o lucro.

Os jogos digitais, hoje, conseguem apropriar-se melhor das referências culturais, sociais e geográficas dadas pela tecnologia e isso tem criado, também, novas formas de comunicação e, principalmente, de agenciamento do jogador, que, ao mesmo tempo em que se diverte, aprende sobre novas culturas, histórias e relacionamentos socioculturais.

A cidade não foi representada tal qual ela é, mas o jogador que vive na cidade ou tem conhecimento de como ela é de fato, achará muitas referências em seu simbolismo [14] e suas associações serão outras, à medida que ele passa por cada uma das fases do jogo. Mas ao mesmo tempo, é possível criar relações com elementos da cultura local e as ações no jogo, principalmente na fase em que nos debruçamos nesse trabalho, onde Max adentra a delegacia de polícia e encontra diversos pratos de coxinha de frango pelas mesas e mobiliários.

Fica evidente que uma busca minuciosa na relação que esse prato, tipicamente paulistano, tem com a polícia militar de

São Paulo, de tal modo que vemos a imagem, fica clara a ligação e a identificação da gíria na cidade de. O policial é mais um coxinha, dentro da trama do jogo e essa associação só pode ser entendida por um interpretante local.

De certa forma podemos perceber uma transição quase que de forma líquida, de um signo que transcende o imaginário local e passa a povoar o ambiente digital, lembrando que esse jogo, por mais que seja ambientado na cidade de São Paulo, faz parte de uma cultura digital, que está em constante transformação e, ao mesmo tempo, faz uso das redes digitais, como [10] nos lembra, e faz uso do hiperespaço transpondo linguagem e lugares.

Não obstante a toda análise para entendimento inicial desse presente artigo, se fez necessário compreendermos o percurso histórico de dois elementos-chaves, antes mesmo de fazermos uma análise sobre o jogo, que foram a coxinha de frango e a polícia militar de São Paulo. Para concluirmos melhor essa análise, temos de nos apoiar novamente em [11], para que essa pesquisa, possa passar uma melhor compreensão sobre nosso estudo.

Observando a história da coxinha de frango, podemos perceber que ela não é apenas o objeto, mas sim o signo em si. Como iguaria paulistana ela quando mencionada remete a sabores, aromas e prazeres, proporcionado pela culinária e que, ao ser lembrada, causa-nos uma certa sensação de prazer, mas que ao mesmo tempo, é apresentada no jogo de modo simbólico, traçando a relação da polícia com a comida pelos motivos já apresentados aqui anteriormente.

A polícia militar no jogo, também é representada e como podemos perceber, seu uniforme, já apresenta que se trata de uma autoridade pública, mas que na história do jogo se mostra com características que não pertencem às corporações policiais, com certos aspectos de mais ligados ao mundo do crime e foras da lei, do que defensor do cidadão e da lei em si.

Emprestando novamente os trabalhos feitos por [11], essa referência visual da roupa, distintivos, maneiras de comportamento e até mesmo ações dentro do jogo, nos fazem perceber, que o Legi-Signo da polícia está completo, emprestando as convenções gerais associados a polícia, como já mencionado, para garantir essa interpretação com certa facilidade por parte do jogador.

Até esse momento podemos entender os dois elementos semióticos com certa clareza, mas ainda nos faltou compreender essa associação entre os dois elementos, coxinha de frango e polícia militar, para só então, compreendermos melhor o simbolismo desses elementos à cultura popular e a partir de agora, validarmos o aparecimento da coxinha de frango durante o jogo.

A relação criada pela coxinha de frango e a polícia, passa a ser simbólica, pois nenhum dos dois signos tem relações diretas, como podemos perceber e isso, só pode ser criado apenas pelo interpretante, que de algum modo, pode ter

contato com essa referência, para criar esse entendimento. Assim como associamos a pomba branca, ligada a uma imagem de paz universal, dada em grande parte, pelos meios de comunicação, essa associação se deu em meio cultural.

Sobre nosso objeto de estudo, aqui apresentado, o que podemos concluir é que essa simbologia não é universal, mas apenas local, pois para que o símbolo seja reconhecido, ela necessita de uma interpretação mais cuidadosa, mas que mesmo assim, pode ser percebida pela produtora, RockStar, antes da produção do jogo, para que pudesse, mesmo que de um modo superficial, pudesse ser representada pelos dois signos e que pudesse criar um simbolismo, já trabalhado por [6] e que nos auxiliasse em seu entendimento.

Portanto, a coxinha de frango sai do prato principal e pula, para um coadjuvante importante na trama do jogo, mesmo que por pouco tempo, mas chamando a atenção daquele que mais nos interessa, o jogador de vídeo game, que já vivencia a experiência de encarar a cidade de São Paulo na sua realidade, não romantizada pelos jogos digitais.

PERSPECTIVAS FUTURAS

Como este artigo trata apenas de um elemento que saltou aos olhos, durante a jornada do jogo, fica evidente que há mais dados que podem ser melhor estudados. Nos cercamos de algumas análises quanto a coxinha de frango, mas ela não é o único elemento semiótico, ali presente, que pode retratar bem a relação da polícia com a sociedade e as implicações culturais no jogo.

Há necessidade de ampliarmos esse olhar para todo o jogo, assim, com melhor atenção, podemos tecer uma análise crítica aprofundada e norteada pela semiótica, para que possamos ter melhores resultados e principalmente, uma melhor compreensão da importância dos jogos digitais para expansão de cultura, além do entretenimento, para um número ainda maior de jogadores e principalmente, a comunidade científica que estuda jogos digitais.

REFERENCES

1. CAVAZIN, Nadir. Histórias e receitas – Sabor, tradição, arte, vida e magia. Sociedade Pró Memória. Limeira. 2000.
2. CASCUDO, L. da C. (Org.). Antologia da alimentação no Brasil. Rio de Janeiro: Livros Técnicos e Científicos, 1977. 254 p
3. LUIZ, Ronilson de Souza. O currículo de formação de soldados da polícia militar frente às demandas democráticas. PUC-SP. Dissertação de Mestrado. Defendido em 2003 – PUC-SP
4. FOUCAULT, Michel. Vigiar e punir : nascimento da prisão. trad. Raquel Ramalhe. 42.ed 1. reimpr. Petropolis, RJ : Vozes, 2014.

5. LIMA, Carlos W. F. de. A construção da imagem realista em jogos de videogame. Um estudo sobre as representações imagéticas nos jogos de nova geração. 2008. Dissertação de Mestrado em Comunicação e Semiótica. Pontifícia Universidade Católica de São Paulo. São Paulo. 2008.
6. LIMA, Leonardo Souza. A Produção de subjetividade nos jogos eletrônicos. São Paulo. Dissertação de Mestrado. Defendida 2011.
7. JENKINS, Henry. Cultura da convergência. São Paulo: Aleph, 2009.
8. JENKINS, Henry. GRENN, Joshua. FORD, Sam. Cultura da Conexão. Criando Valor e significado por meio da mídia programável. São Paulo: Aleph, 2015.
9. NEGRI, Antonio. [A nova ordem mundial na era do capitalismo cognitivo](#). 2010. Acessado em 20/12/2015: <http://osertanejo.multiply.com/photos/hi-res/1M/109>.
10. SANTAELLA, Lucia. Comunicação Ubiqua: Repercussões na cultura e na educação. 2007
11. PEIRCE, Charles Sanders. Semiótica. Ed. Perspectiva. 2000.
12. SANTAELLA, Lucia, FEITOZA, Mirna. Mapa do Jogo: A diversidade cultural dos games. São Paulo. Cengage. 2009.
13. CARNEIRO, Henrique S. Comida e sociedade: significados sociais. Na história da alimentação. História questões e Debates, Curitiba. Editora UFPR, 2005
14. SALEN, Katie ZIMMERMAN, Eric. Regras do jogo, fundamentos do design de jogos Vol1, 2, 3, 4. São Paulo. Blucher. 2012.
15. STUMPF, Andre Gustavo. MERVAL, [Pereira Filho](#). A segunda guerra sucessão de Geisel. Ed. Brasiliense. São Paulo. 1979. Disponível em: <http://www.significados.com.br/cozinha/> acessado em 04 de agosto de 2015.

JOT: A Modular Multi-purpose Minimalistic Massively Multiplayer Online Game Engine

Gonçalo N. P. Amador

Instituto de Telecomunicações
Covilhã, Portugal
g.n.p.amador@gmail.com

Abel J. P. Gomes

Universidade da Beira Interior
Covilhã, Portugal
agomes@di.ubi.pt

ABSTRACT

Most game engines are developed by and to the game industry. They are normally *monolithic*, i.e., typically grown out of a specific game, tendentiously gender specific, whose components (e.g., physics sub-engine) cannot be considered as separate modules, i.e., their functionalities are not clearly separated from each other. Consequently, there is a lack of *modular*, well documented, open source frameworks for games. In this paper, we present JOT, a minimal, modular game engine for Massively Multi-player Online Games (MMOGs); recall that ‘jot’ means minimal thing.

Author Keywords

Game Engine; Educational Software Tools; MMOGs.

INTRODUCTION

In game industry, game engines are used to produce games. Their goal is to reduce game development time through a collaborative process between programmers and artists. Game engines allow that both programmers and artists work separately or simultaneously in a given project. Historically, game engines evolved into two major variants, namely: *monolithic* and *modular*. *Monolithic* game engines evolved from specific games in order to allow content and behavior changes, via scripting (e.g., Unreal Engine [30]). *Modular* game engines are a set of tools for either game gender that, in theory, give freedom to alter the source code or replace or modify its core components (e.g., render sub-engine, physics sub-engine, AI sub-engine, etc.). Other engines (e.g., Unity3D [6]), built for multi-cross platform purposes, are in fact game engines to built *monolithic* game engines [21]. Regardless, former core sub-engines can not be replaced; instead, they can be

extended with new features, i.e., they are partially *modular* game engines [21]. The main contribution of this paper is JOT, a modular game engine for research and teaching of game engine technologies and MMOGs [1].

The remain of this paper is organized as follows. Section 2 overviews game engines. Section 3 briefly describes JOT architecture. Section 4 details JOT *infrastructure layer*. Section 5 details JOT *core layer*. Section 6 details JOT *toolkits layer*. Section 7 details JOT *framework layer*. Finally, Section 8 draws relevant conclusions and points out new directions for future work.

GAME ENGINES OVERVIEW

Game Engines: the Past

The first game engines appeared in the 1980’s (e.g., ASCII’s RPG Maker [11]), in particular for 2D games. In the 1990’s, with the launch of general purpose application programming interface (APIs), for rendering 2D and 3D vector graphics (i.e., OpenGL and DirectX), game developers began using higher level languages. More specifically, in the 1993, id Software “Doom engine” became the first commercially available 3D game engine [4, 26]. It was the first engine to structure the components of computer games (e.g., rendering, A.I., physics, networking, etc.). This allows for using the same game engine to develop other games with other contents. Regardless, until the end of the 1990’s game engines where mostly focused on a specific game genre (e.g., First-Person Shooters (FPS)) [4, 26].

After 2000, game engines development went in various directions, namely:

1. Multiple game genders and platforms, e.g., Unity3D [6].
2. Mobile devices, e.g., wGEM [28], M3GE [15], mOGE [19], etc.
3. Teaching/studying game technologies, e.g., Gedi [8], Minueto [9], enJine [23], Mammoth [17], etc.

4. Design of novel game engines from scratch with arguably novel architectures, e.g., CryEngine [22] or [Amazon Lumberyard](#).
5. Porting of game engines components (e.g., render module) functionalities into the GPU, e.g., via shader languages [18].

Game Engines: the Present

At the present, game engines are in their vast majority complex tools, often permitting to recode portions of the engine core or the modify it via a general (e.g., Lua) or specific (e.g., UnrealScripting) scripting language. Some game engines strive for being suited for multiple genders and multiple platforms, e.g., Unity3D [6]. Note that many game engines developed in academia for a specific field of study are either no longer open source [13, 23, 3, 7, 16, 17] or abandoned [9]. Nevertheless, open source game engines are too complex or not enough modular, so that the removal or rewriting of its components in a difficult task. But, recall that most major game companies use proprietary game engines.

Game Engines: the Future

We have identified five major future research trends in the development of game engines, namely:

1. The replacement of traditional raster-based renderers by ray tracing-based renderers [31]. In the render module of game engines and other graphical applications, ray casting (i.e., a particular ray tracer limited to primary rays) has been used in game development for many years [12], mainly for collision detection. This is so because it is already feasible to run a ray tracer in real-time for not so much complex 3D scenes using shader programming or more general programming like CUDA or OpenCL, so that the expectations are to have ray tracers with real-time performance in the near future [25].
2. The introduction of voxelized scenes for scene management, including physics and complex terrains. This particularly useful for handling terrain features like caves or overhangs, as well as destructive war scenarios with falling off buildings and the like.
3. The development of automated procedural generation algorithms. One of the major problems of current games is the construction of scenarios for game levels, though most game engines include scene editors. In the future, we will see automated generation of 3D scenes from a descriptive specification in order to shorten the process of game prototyping and development.
4. GPU-centric game engines. At present, most parallel features of game engines are provided by shaders, particularly for illumination purposes, though some efforts were made to integrate commercial games with NVIDIA CUDA (e.g., Mirror's Edge [24]). Therefore, in the future, we will assist to the development of the game engines that fully leverage the processing power of GPUs.
5. The emergence of really modular game engines. Nowadays, most game engines do not permit the replacement of a game core component easily. In fact, there is no known game engine or framework that allows for the replacement of any sub-engine (e.g., physics, A.I., networking, etc.) by another one; for example, it is not possible to replace a physics sub-engine (e.g., Havok) by another one (e.g., Bullet) in a straightforward manner. It is clear that this would be feasible if there was a common interoperability framework for physics sub-engines.

JOT architecture takes into consideration the former aspects to attempt to ensure modularity and future extensibility.

JOT ARCHITECTURE

JOT is built in Java programming language for several reasons, namely:

1. The Java Virtual Machine (JVM) is multiplatform.
2. There are considerable third-party Java application layer multicast implementations (e.g., [JGroups](#)) and cloud/grid middlewares (e.g., [GridGain](#)).
3. There is a significant base of game engines for teaching computer graphics courses and/or studying game technologies in Java, e.g., M3GE [15], Minueto [9], enJine [23], Mammoth [17], etc.
4. There are open source game engines in Java that serve as a base for design decisions and performance testing, e.g., [jMonkeyEngine](#) and [Jake2](#), an open source port to Java of Quake II.

JOT gathers many concepts of other game engines developed in academia. First, it has a layer structure similar to that one of RAGE [20]. Second, it uses concepts implemented in M3GE and jMonkeyEngine scene management, i.e., its classes and methods are similar to those found in M3GE and jMonkeyEngine. Third, the organization of toolkits of JOT is strongly influenced by Mammoth; e.g., JOT *artificial intelligence toolkit* uses a similar design as Mammoth AI module. JOT is a four-layer game engine, from bottom to top: *Infrastructure*, *Core*, *Toolkits*, and *Framework*, as illustrated in Fig. 1.

As observed in Fig. 1, dashed rounded corner rectangles are external libraries, e.g., *JOGL*. Any layer component that resorts to an external library obeys to interface/abstract classes as expected by above layers, e.g., if we replace JGroups with [Orbit](#), solely the *Network Toolkit* must be re-implemented obeying to the same interface/abstract classes, without any changes required in other toolkits or layers. Each layer constituents can only use functionalities of the same layer or lower layers constituents, e.g., the *Artificial Intelligence Toolkit* relies on the *Geometry Extended Toolkit*, on the *Core layer*, and on the *Infrastructure layer*. In order to replace a component in the engine, one might solely adapt the respective toolkit or additionally modify its lower layers, e.g., to replace JOT physics code with a physics engine one must implement classes that obey to the provided interface/abstract classes in the *Physics Extended Toolkit* and, if desired, in physics *Core layer*. This implies that modularity of the game engine is ensured at the expense of possible redundancy or unused source code in the *Core layer*.

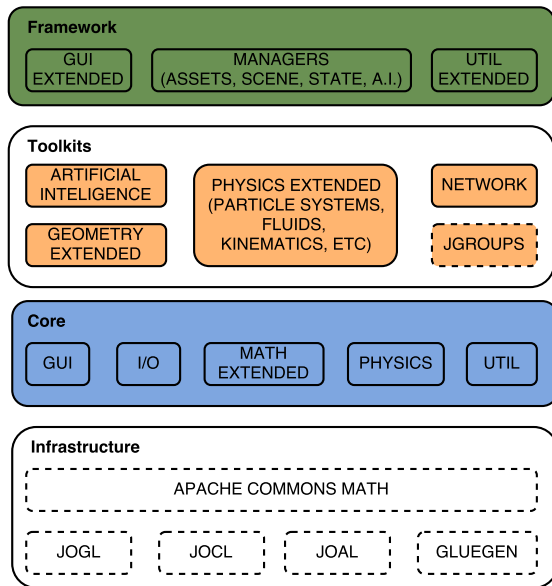


Figure 1. JOT layers.

INFRASTRUCTURE LAYER

This layer is comprised of five libraries: *The Apache Commons Mathematics Library*, *JOGL*, *JOCL*, *JOAL*, and *GLUEGEN*, with the latter four wrapped in *JogAmp*. Note that JogAmp provides the graphical pipeline, sound, and OpenCL support. Presently, JOT uses JOGL for the graphical pipeline (OpenGL) and for 2D/3D sound support (OpenAL). Recall that there is not a Java binding for DirectX. *The Apache Commons Mathematics Library* is an extensible math library for general purpose scientific use, chosen by its extensive features and continuous support/development, in detriment of Java3D *Vecmath*, used for example in jMonkeyEngine. However, many of the *The Apache Commons Mathematics Library* functionalities are implemented for doubles and not for floats, thus some upper layers of JOT re-implement some of its functionalities for single precision floating-point numbers instead. More libraries, namely, OpenCV for camera or control devices, such as the WiiMote, might also be added in this layer.

CORE LAYER

Core is the minimal set of tools that can allow the classification of a piece of software as a game engine, i.e., games can be written on the top of the *Core* layer, with the assumption that we have the infrastructure layer underneath. The *Core* layer is comprised of five modules, as detailed below.

GUI Module

This module regards a simple game graphical application with the following functionalities: initialize the game, i.e., load the menus/controls devices and show startup info; load all default game content, i.e., player, obstacles, SkyBox, and floor; running the game main thread; shutdown the game, clean up code that runs only once; the loop that runs until the game ends, that continuously and as many times as possible

per second processes the input handlers events, update the state and position of all the game objects, detect collisions and provide responses, and refreshes the display. Some of this module abstract single class *SimpleGame.java* methods are guidelines (abstract methods) to be implemented in a game project, i.e., an game main class extends this class.

Math Extended Module

This module is built upon *The Apache Commons Mathematics Library*. It regards geometry in a graphical scene. It contains *shape* class and its subclasses for triangles, planes, and spheres, which are the constituents of a *mesh* class. Note that each mesh object (and its bounding volume(s)) can be associated to a *transform group* for scene modeling purposes, as usual in scene graphs. Implemented *bounding volumes* are dynamic axis aligned bounding box (AABB), object aligned bounding box (OBB), and *bounding sphere*. A dynamic AABB changes size depending on the mesh rotation to ensure that all mesh vertices are within the AABB. Obviously, the abstract classes for shapes and bounding volumes are written to accommodate new subclasses for specific shapes and bounding volumes, i.e., they can be extended. Also, this module implements single and double precision real numbers, linear and cubic interpolation, and geometric intersection algorithms, as needed in collision detection, though only the Gilbert-Johnson-Keerthi (GJK) distance algorithm has been implemented [14].

I/O Module

This module is where the 2D/3D sound, scripting engine, control devices, and geometric loaders are implemented. The geometric loaders currently support the Collada 1.4.1 format (i.e., geometry and material data), but Wavefront OBJ and MD2 quake 2 files formats are not fully supported. The sound is managed by an handler for 2D/3D sound, which was implemented on top of OpenAL. The scripting engine serves the purpose of loading configuration options in the *Util* module (more specifically, in *CoreOptions* class), though this scripting engine might be extended to incorporate other functionalities. Note that the I/O module is designed in order to be extended with classes or libraries to interpret scripting languages (e.g., Lua or Phyton), and other control devices aside mouse or keyboard, such as WiiMote or Kinect. The control devices classes implement one or two interfaces, namely:

1. *Input* is the interface for all control devices that do not require their positions to be tracked, e.g., keyboard, joystick, steering wheel, WebCam, Kinect, etc.
2. *TrackableInput* is the interface for all control devices that require tracking of their positions or acceleration, e.g., mouse, Wiimote, etc.

These two interfaces require the registration of user-interaction events, e.g., a key or button is pressed in a specific position. This is achieved using the *isDetecting* or *isContinuouslyDetecting* methods. Trackable controls are those that have gyroscopes and/or accelerometers.

Physics Module

The physics module consists of the following interfaces/classes: *material* (for objects), *ray* (of light), *intersectionResult* (between ray and objects). Note that the *mesh* class has a list of materials. The *ray* and *intersectionResult* classes are for upper layer support for ray casting [31] or ray tracing [12] implementations.

Utils Module

This module specifies the options (boolean values) of the core, and implements a generic class called *CoreGameObject*. The *GameObject* class is an extension to the abstract class *CoreGameObject*, which in turn extends the *TransformGroup* class. A *GameObject* is the most generic type of game object, no matter the the game gender; this class includes not only data concerning geometry and materials, but also data related with speed, maximum speed, attributes, and so on.

TOOLKITS LAYER

This layer includes toolkits, which are extensions to the core. There are toolkits for a number of purposes, namely artificial intelligence (AI), geometry generators, physics simulation, etc. Each toolkit includes an *Utils* component which specifies its options (boolean values).

Geometry Extended Toolkit

This toolkit extends the features of core geometry module, with: *terrain generators*, via fractal or Brownian motion noise generators; *terrain smoothing* algorithms, via parametric surfaces or Gaussian blur/smooth; *maze generation*, using Prim's algorithm; *quadrics*, *SkyDome*, and roughly spherical *celestial objects* (planets, moons, and stars).

Physics Extended Toolkit

This toolkit provides several new features to the core physics, namely: *collision handling* among rigid (either moving or not) bodies; *kinematics* for moving bodies and for projectiles; *particle system effects*, for explosions and 2D fluids; *Euler 2D fluids*; *hybrid* (Euler and particle fluid physics) or *procedural* rain ripply effects. As a margin note, work is undergoing to extend all these 2D physics effects to 3D.

Artificial Intelligence (AI) Toolkit

The *AI Toolkit* supports steering behaviors [29], where each behavior implements a common *steering behavior* interface. The implemented steering behaviors are, *seek*, *pursuit*, *flee*, *evade*, *arrive*, *collision avoidance*, *interpose*, *wander*, *path following*, *offset pursuit*, and *flocking behavior*. It also supports pathfinding algorithms, where each pathfinder implements a *pathfinder* interface. The implemented pathfinders include Dijkstra's algorithm [10], A* [2], Fringe search [5], and Best-First search [27].

Network Toolkit

This toolkit implements the generic network communication of JOT. It implements an interface that serves to abstract communication using any networking API (e.g., JGroups and GridGain), regardless of its network topology. Therefore, if the networking API was replaced by another for some reason, game implementations that resort to this toolkit will still work.

FRAMEWORK LAYER

This is the upper layer of JOT. This layer comprises three modules: GUI, managers, and Util Extended. This layer aims at the following: first, to provide management of the application/game state and scene; second, to separate the game logic from its graphical application. As a margin note, a partially implemented off-line ray-tracer is integrated within this layer.

GUI Extended Module

This module provides the following classes: *FramePerSecond*, *Game* (an abstract class that every single game must extend), and *Camera* (another abstract class). This latter class has the following sub-classes: *FirstPerson* for first-person shooters, *ThirdPerson* for role-playing games, and *StaticPerspective*, *TrackingGameObjectPerspective*, *StaticUpperView* and *TrackingGameObjectUpperView* for strategy games.

Manager Module

This module is responsible for scene, state, and AI managers are implemented. The scene manager deals with handling of the scene via a tree-based scene graph, permits the usage of planar or shadow volume-based shadows, and handles collisions among objects within the scene graph. The state manager handles game state, i.e., information of each game object (e.g., life, speed, etc.), no matter we are using a client-server network topology or a peer-to-peer network topology; which is particularly useful for massively multiplayer online games (MMOGs). Finally, there are two AI managers: the first for steering behaviors and the second for path-finding.

Util Extended Module

This module implements a specification of options (boolean values) of the *Framework* layer for usage in the game application/implementation.

CONCLUSIONS AND FUTURE WORK

JOT current architecture is satisfactory in proposed features and modularity. However, JOT still lacks several features and improvements to existent features, namely:

1. Extend the *I/O* module, to fully support Wavefront OBJ and MD2 quake 2 files format, and Collada 1.4.1 or 1.5 skeleton animation.
2. Extend the Euler and particle fluids to 3D.
3. Replace JOT toolkits with third-party solutions, e.g., replace the *physics extended toolkit* with the [JBullet](#) physics engine.
4. Improve JOT in order to make it a common interoperability framework or interface among the many game related middlewares for many tasks, physics and render to name a few.
5. Parallellize JOT in order to make it a general purpose parallel game engine, using OpenCL. To be clear, we do not refer solely to using shaders mostly for illumination effect and/or render. Instead the idea it to run most of the game engine features inside the graphics card, e.g., a Ray Caster [12], e.g., for collision detection, and a Ray Tracer [31] for rendering of the game.

6. Improve JOT in order to make it a common interoperability framework or interface among the many game related middlewares for many tasks, physics and render to name a few.
7. Support JOT with cloud technologies, i.e., make it able to be used by multiple worldwide distributed game developers.
8. Allow games to be rendered either directly on the browser or as an application.
9. Modify JOT in order to allow it to be used in the creation of mobile applications for Android.

As priorly stated a few of these improvements are already undergoing work.

ACKNOWLEDGMENTS

This research has been partially supported by the Portuguese Research Council (Fundação para a Ciência e Tecnologia), under the doctoral Grant SFHR/BD/86533/2012, and also by FCT Project UID/EEA/50008/2013.

REFERENCES

1. G. Amador and A. Gomes. 2016. A Video Games Technologies Course: Teaching, Learning, and Research. In *EG 2016 - Education Papers*. The Eurographics Association.
2. A. Bagchi and A. Mahanti. 1983. Search Algorithms Under Different Kinds of Heuristics—A Comparative Study. *J. ACM* 30, 1 (1983), 1–21.
3. J. Bernardes, R. Nakamura, D. Calife, D. Tokunaga, and R. Tori. 2009. Integrating the Wii Controller with enJine: 3D Interfaces Extending the Frontiers of a Didactic Game Engine. *Computers in Entertainment (CIE)* 7, 1 (2009), 1–19.
4. L. Bishop, D. Eberly, T. Whitted, M. Finch, and M. Shantz. 1998. Designing a PC Game Engine. *IEEE Computer Graphics and Applications* 18, 1 (1998), 46–53.
5. Y. Björnsson, M. Enzenberger, R. Holte, and J. Schaeffer. 2005. Fringe Search: Beating A* at Pathfinding on Game Maps. In *Proceedings of the 2005 IEEE Symposium on Computational Intelligence and Games (CIG '05)*, Essex University, Colchester, Essex, UK, 4-6 April, 2005. IEEE Computer Society.
6. S. Blackman. 2011. *Beginning 3D Game Development with Unity: All-in-one, Multi-platform Game Development* (1st ed.). Apress, Berkely, CA, USA.
7. Jean-Sébastien Boulanger. 2006. *Interest Management For Massively Multiplayer Games*. Master's thesis. School Of Computer Science, McGill University, Montréal, Canada.
8. R. Coleman, S. Roebke, and L. Grayson. 2005. Gedi: a game engine for teaching videogame design and programming. *Journal of Computing Sciences in Colleges* 21, 2 (2005), 72–82.
9. A. Denault. 2005. *Minueto, an Undergraduate Teaching Development Framework*. Master's thesis. School Of Computer Science, McGill University, Montréal, Canada.
10. E. Dijkstra. 1959. A Note on Two Problems in Connexion with Graphs. *NUMERISCHE MATHEMATIK* 1, 1 (1959), 269–271.
11. M. Duggan. 2011. *RPG Maker for Teens* (1st ed.). Course Technology Press, Boston, MA, United States.
12. J. Ellis, G. Kedem, T. Lierly, D. Thielman, R. Marisa, J. Menon, and H. Voelcker. 1991. The Ray Casting Engine and Ray Representatives. In *Proceedings of the First ACM Symposium on Solid Modeling Foundations and CAD/CAM Applications (SMA '91)*. ACM Press, 255–267.
13. L. Emmerich, D. Tanaka, R. Petriche, F. Kamakura, and J. Bernardes. 2006. Building the Network Module for a Didactic Game Engine. (2006).
14. E. Gilbert, D. Johnson, and S. Keerthi. 1988. A fast procedure for computing the distance between complex objects in three-dimensional space. *IEEE Journal of Robotics and Automation* 4, 2 (1988), 193–203.
15. P. Gomes and V. Pamplona. 2005. M3GE: um motor de jogos 3D para dispositivos móveis com suporte a Mobile 3D Graphics API. In *Proceedings of the IV Brazilian Symposium on Computer Games and Electronic Entertainment*. Sociedade Brasileira de Computação (SBC), 55–65.
16. M. Hawker. 2008. *Subgames in massively multiplayer online games*. Master's thesis. School Of Computer Science, McGill University, Montréal, Canada.
17. J. Kienzle, C. Verbrugge, B. Kemme, A. Denault, and M. Hawker. 2009. Mammoth: a massively multiplayer game research framework. In *Proceedings of the 4th International Conference on Foundations of Digital Games (FDG '09)*. ACM Press, 308–315.
18. A. Lefohn, M. Houston, J. Andersson, U. Assarsson, C. Everitt, K. Fatahalian, T. Foley, J. Hensley, P. Lalonde, and D. Luebke. 2009. Beyond Programmable Shading (Parts I and II). In *ACM SIGGRAPH 2009 Courses*. ACM Press, 1–312.
19. I. Macêdo, Jr. 2005. *mOGE - mObile Graphics Engine: O projeto de um motor gráfico 3d para a criação de jogos em dispositivos móveis*. Final report. Universidade Federal de Pernambuco (UFPE), Pernambuco, Brazil.
20. L. McCulloch, A. Hofman, J. Tulip, and M. Antolovich. 2005. RAGE: A Multiplatform Game Engine. In *Proceedings of the Second Australasian Conference on Interactive Entertainment (IE '05)*. Creativity & Cognition Studios Press, 129–131.
21. F. Messaoudi, G. Simon, and A. Ksentini. 2015. Dissecting Games Engines: The Case of Unity3D. In *Proceedings of the 2015 International Workshop on Network and Systems Support for Games (NetGames '15)*. IEEE Press, 4:1–4:6.

22. M. Mittring. 2007. Finding Next Gen: CryEngine 2. In *ACM SIGGRAPH 2007 Courses*. ACM Press, 97–121.
23. R. Nakamura, L. Bernardes, and R. Tori. 2006. enJine: Architecture and application of an open-source didactic game engine. In *Proceedings of the Digital V Brazilian Symposium on Computer Games and Digital Entertainment (SBGAMES'2006)*.
24. J. Norton, C. Wingrave, and J. LaViola, Jr. 2010. Exploring Strategies and Guidelines for Developing Full Body Video Game Interfaces. In *Proceedings of the Fifth International Conference on the Foundations of Digital Games (FDG '10)*. ACM Press, 155–162.
25. S. Parker, H. Friedrich, D. Luebke, K. Morley, J. Bigler, J. Hoberock, D. McAllister, A. Robison, A. Dietrich, G. Humphreys, M. McGuire, and M. Stich. 2013. GPU Ray Tracing. *Commun. ACM* 56, 5 (2013), 93–101.
26. P. Paul, S. Goon, and A. Bhattacharya. 2012. HISTORY AND COMPARATIVE STUDY OF MODERN GAME ENGINES. *International Journal of Advanced Computer and Mathematical Sciences* 3, 2 (2012), 245–249.
27. J. Pearl. 1984. *Heuristics: Intelligent Search Strategies for Computer Problem Solving*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.
28. A. Pessoa, G. Ramalho, and A. Battaiola. 2002. wGEM : Um Framework de Desenvolvimento de Jogos para Dispositivos Móveis. In *Proceedings of the XXIX Seminário Integrado de Software e Hardware*. Sociedade Brasileira de Computação (SBC).
29. C. Reynolds. 1999. Steering Behaviors For Autonomous Characters. (1999). Retrieved September 16, 2016 from <http://www.red3d.com/cwr/papers/1999/gdc99steer.pdf>.
30. L. Surhone, M. Tennoe, and S. Henssonow. 2010. *UnrealScript*. Betascript Publishing, Mauritius.
31. T. Whitted. 1980. An Improved Illumination Model for Shaded Display. *Commun. ACM* 23, 6 (1980), 343–349.

Arquiteturas de rede em jogos multi jogador no Unity

Nuno Carapito

Universidade da Beira Interior
Covilhã, Portugal
nfac93@gmail.com

Frutuoso G. M. Silva

Instituto de Telecomunicações
Universidade da Beira Interior, Covilhã, Portugal
fsilva@ubi.pt

RESUMO

O desenvolvimento de jogos digitais tem vindo a aumentar em todo o mundo, sendo estes desenvolvidos por empresas, pessoas individuais ou pessoas organizadas em pequenas equipas. Estes últimos, normalmente, contam com o apoio de motores de jogo, que têm como principal objetivo acelerar o desenvolvimento de jogos.

Nos jogos digitais, a componente multi jogador tem vindo a ganhar uma importância enorme, sendo atualmente raro um jogo sair para o mercado sem ter esta componente.

Este artigo apresenta a comparação entre as arquiteturas *peer-to-peer* e cliente/servidor usadas no desenvolvimento de um jogo multi jogador. Para este teste foi desenvolvido um protótipo usando o *Unity* para verificar qual das duas arquiteturas de rede é melhor para um jogo de estratégia multi jogador. Os testes preliminares realizados com jogadores, indicaram que a arquitetura *peer-to-peer* parece funcionar melhor.

Palavras Chave

Jogos digitais; multi jogador; arquitetura cliente-servidor; arquitetura peer-to-peer; motor de jogo; *Unity*.

INTRODUÇÃO

Nos dias que correm qualquer pessoa pode desenvolver um jogo, visto que existe uma quantidade elevada de tutoriais na Internet, uma documentação bastante completa e uma comunidade que ajuda imenso. No entanto, existe falta de documentação sobre como implementar diferentes arquiteturas de rede para jogos multi jogador *online*, e, entre as diferentes arquiteturas, quais as vantagens de umas comparativamente às outras, bem como saber qual delas se deve implementar para o tipo de jogo desejado. Este trabalho tenta colmatar essa falha

de informação, orientada aos jogos de estratégia multi jogador desenvolvidos em *Unity*

Problema e Objetivos

O desenvolvimento de jogos digitais tem vindo a encontrar diversas barreiras ao longo dos anos, o que por vezes impediram o lançamento dos mesmos. Um exemplo disto é o *World Of Warcraft*, que encontrou uma barreira muito grande no lançamento do jogo a nível de arquitetura de rede, uma vez que cada servidor aguentava no máximo cinco mil jogadores ao mesmo tempo, tendo sido criados centenas de servidores diferentes. Com o passar do tempo, com maior capacidade de processamento por parte do *Hardware*, este limite tem sido aumentado e os servidores já chegam a ter cerca de dez mil jogadores e muitos deles interligados entre si, podendo os jogadores de um servidor jogar com jogadores de outro servidor. No entanto, continua a ser usada uma arquitetura cliente/servidor, que traz elevados custos para a empresa, que tem de sustentar toda a infraestrutura. Nos últimos anos tem sido feito um esforço a nível global para arranjar forma de transformar este tipo de arquitetura numa arquitetura *peer-to-peer*, na qual a informação é trocada diretamente entre os clientes, aliviando assim os encargos das empresas com os servidores.

O principal objetivo deste projeto é verificar qual das duas arquiteturas funciona melhor em jogos de estratégia multi jogador. Para isso, foi implementado um pequeno protótipo em *Unity* usando os dois tipos de arquitetura: a cliente/servidor e *peer-to-peer*. Um fator crítico no desenvolvimento deste protótipo foi o facto de não existir muita documentação do *Unity* sobre o desenvolvimento de jogos multi jogador *online*, usando qualquer uma das duas arquiteturas.

JOGOS MULTI JOGADOR

Os jogos multi jogador são jogos nos quais duas ou mais pessoas podem jogar no mesmo ambiente e ao mesmo tempo. Estes jogos permitem uma interação entre jogadores, quer seja em parceria ou em competição, oferecendo uma componente social que não existe nos jogos de um só jogador. Este tipo de jogos pode funcionar em dois modos distintos: o modo local e o modo *online*.

Em modo *online*, os jogos requerem que os jogadores se liguem a um servidor central (independente da arquitetura de rede usada), de forma a conseguir jogar com outros jogadores, utilizando uma ligação à Internet. Em modo local, os jogadores podem usar *Local Área Network (LAN)/Wireless Local Area Network(WLAN)*, ou até mesmo *Bluetooth*, para criar uma rede local com outros jogadores, não sendo necessária uma ligação à Internet.

O Aparecimento dos Jogos Multi Jogador

Os jogos multi jogador tiveram um aparecimento discreto, uma vez que os primeiros eram simples jogos só para um jogador, mas com características que eram visíveis em comunidades, como por exemplo, listas de melhores pontuações, torneios e *chats*. Isto permitia aos jogadores comparar os seus resultados com outros jogadores, bem como competir para ver quem conseguia obter mais pontos. Este tipo de jogos foram apelidados de “jogos competitivos orientados à comunidade” [1].

Os jogos multi jogador apareceram há mais de quarenta anos. Uns dos primeiros foram o *Empire* de 1973 e o *Spasim* de 1974, suportando oito e trinta e dois jogadores, respetivamente. Antes destes foram desenvolvidos jogos multi jogador, mas com um limite máximo de dois jogadores, como por exemplo o *Tennis for Two* e o *SpaceWar!*. No entanto, os recursos do jogo tinham de ser partilhados por todos os jogadores. Anos mais tarde, começaram a ser lançados jogos com base numa ligação de rede, como por exemplo o *Midi Maze*. Após estes, começaram a aparecer os jogos mais conhecidos, como por exemplo o *Doom*, um *First-Person Shooter*, com dois estilos de jogo, *Deathmatch* e *Arena*. Posteriormente, os mais conhecidos a surgir foram o *Counter-Strike*, o *Halo* e o *Unreal Tournament*.

Hoje em dia, o desenvolvimento de jogos com uma componente multi jogador em tempo real é cada vez mais popular.

Arquiteturas para Jogos Multi Jogador

Com o avançar do tempo e da tecnologia, a quantidade de jogos multi jogador tem vindo a crescer, não só em popularidade mas também em escala. Existem duas grandes arquiteturas de redes de computadores para jogos multi jogador: a arquitetura cliente/servidor e a arquitetura *peer-to-peer*.

Atualmente, na maioria dos jogos multi jogador é usada a arquitetura cliente-servidor. Neste tipo de arquitetura existe um servidor central ao qual todos os clientes (ou seja, todos os jogadores) se ligam para poderem começar a jogar. Todas as ações tomadas pelos jogadores são enviadas para o servidor sob a forma de uma mensagem. O servidor processa de forma sequencial as mensagens recebidas e vai responder aos clientes os efeitos das suas ações, também através de mensagens. A comunicação só ocorre entre o servidor e cliente e nunca entre os vários clientes. Este tipo de arquitetura começou a ser usada não só pela maior facilidade de implementação, mas também pela capacidade de evitar que os clientes façam

batota⁰, visto ser o servidor a tomar as decisões mais importantes.

Com o aumento da complexidade dos jogos multi jogador *Massively Multiplayer Online Games (MMOGs)*, está a aparecer um problema grave na arquitetura clássica cliente-servidor (principalmente nos *Massively Multiplayer Online Role-Playing Games, MMORPG*), devido ao elevado custo do preço em *hardware* e *data centers*. Nos últimos tempos, tem sido feita uma pesquisa intensa para adaptar a arquitetura *peer-to-peer* aos *MMOGs*, para espalhar a carga pelas máquinas dos jogadores [2]. Neste tipo de arquitetura, cada um dos jogadores funciona tanto como cliente quanto como servidor, existindo assim uma troca de informação entre os jogadores, sem que haja necessidade de um servidor central. No entanto, só a informação mais importante pode ser trocada com os vários jogadores, caso contrário, vai existir uma quantidade de informação bastante grande para ser trocada entre os vários jogadores, o que pode causar alguns problemas. É uma arquitetura bastante difícil de implementar, para além de também ser muito difícil prevenir que alguns jogadores façam batota, visto que os jogadores com experiência em redes podem alterar a informação dos pacotes de rede que enviam aos outros jogadores, sendo assim beneficiados (tal não é possível na arquitetura cliente/servidor pois é o servidor que faz todos os cálculos).

Principais Problemas dos Jogos Multi Jogador

Segundo Badar et al.[1], os principais problemas dos jogos *online* são: latência, perda de pacotes e pouca largura de banda.

A latência e o limite na largura de banda são os dois principais problemas nos jogos *online*. Relativamente à perda de pacotes, era um grave problema até há uns anos atrás. No entanto, já foram criados métodos para evitar que a perda de um pacote afete uma partida entre vários jogadores.

A largura de banda é definida como a quantidade de dados que pode ser transmitida do ponto A para o ponto B, num determinado período de tempo, sendo chamada na gíria por “limite de tráfego”. Embora nos países mais desenvolvidos as larguras de banda sejam já bastante elevadas, ainda existem países com larguras de banda bastante reduzidas. Para além disso, existe também um limite bastante reduzido nas redes móveis, sendo do interesse do jogador que a quantidade de dados a serem transmitidos seja o mais baixo possível.

Outro problema é a latência, que é o tempo demorado entre o envio de um pedido (pacote de dados) e a chegada da sua resposta. Os jogos em tempo real necessitam de baixas latências, visto que um atraso no tempo de resposta pode causar degradação na experiência de jogo dos utilizadores. *Michael Powers* [3] afirma que o principal problema de um jogo em tempo real é a latência da rede. Um jogo com elevada latência vai afetar os jogadores, levando a um elevado nível de inconsistência, podendo até dar vantagem injusta para alguns deles, removendo assim a igualdade entre todos.

⁰ Jogador altera a informação enviada nos pacotes de rede, ganhando vantagem sobre os outros jogadores.

Segundo Wang et al.[6], os efeitos da latência em jogos *online* podem ser categorizados da seguinte maneira: eficiência da rede, consistência visual, consistência do mundo de jogo e igualdade.

Para resolver o problema de eficiência da rede pode ser usado a técnica de *Content Addressable Network*(CAN), que é um sistema distribuído *peer-to-peer* e descentralizado que mapeia chaves n-dimensionais em valores. O benefício principal de usar esta técnica advém da menor necessidade de pacotes relativamente a uma transmissão simples. Outra solução é gravar em *buffer* as mensagens do estado do jogo, e, em vez de se enviar várias mensagens individualmente, enviar um conjunto delas, causando menos problemas na rede.

Para resolver o problema da consistência visual existe uma técnica chamada *Dead Reckoning Technique* [6]. Normalmente, num jogo em tempo real, existem bastantes objetos, tais como criaturas, veículos, entre outros. Para que todos os jogadores possam receber uma atualização de todos os objetos é necessário transmitir uma quantidade elevada de informação entre o servidor e os clientes. Isto causa um enorme consumo de banda de rede e aumenta o risco de latência. Esta técnica extrapola a posição exata do objeto ao saber a sua última posição e velocidade. Com esta técnica, os clientes apenas precisam de dizer ao servidor as mudanças de velocidade dos objetos sobre o controlo do utilizador.

Relativamente à consistência do mundo de jogo, existe um método chamado de *Bucket Synchronisation Mechanism*, que tem a capacidade de recolher todas as mensagens de eventos do jogo e guardá-las num *bucket*. Em conformidade com um intervalo específico, todas as mensagens no *bucket* são processadas e é criada uma visão local do estado global. Esta é a melhor forma de sincronização para jogos multi jogador *online*, usando o sistema *peer-to-peer* [6].

Relativamente ao último problema, a igualdade é uma parte essencial para que um jogo seja justo e divertido para os jogadores. Uma das formas de resolver este problema é com o método chamado *Sync-MS*, que promove a igualdade nos jogos ao balancear o tempo de resposta. Visto que clientes de todo o mundo experienciam latências variadas, alguns jogadores podem tomar ações de acordo com a última mensagem, antes de outros jogadores terem sequer recebido essa mesma mensagem. Idealmente, todos os clientes devem receber as mensagens de atualização do servidor ao mesmo tempo, o que é conseguido através do mecanismo *Sync-out*. Este mecanismo coloca as mensagens de atualização em linha de espera e aguarda que estas tenham chegado a todos os clientes. Só após todos os clientes terem recebido a mensagem é que ela é entregue ao jogo, o que permite que os jogadores possam reagir ao mesmo tempo.

Sheldon et al.[4] partilharam na sua pesquisa o impacto da latência nos jogos de estratégia, referindo que não existem efeitos perceptíveis quando a latência aumenta até 500 milissegundos. Latências por volta de 800 milissegundos causam degradação da experiência do jogo e transmitem informação errada ao jogador. No entanto, a frequência de atualizações nos jogos de estratégia não precisa de ser assim tão elevada,

pelo que se pode “esconder” a latência, sem que o jogador tenha precepção da mesma.

Motores de Jogo

Um motor de jogo (*Game Engine* em Inglês) é um *framework* desenhado para a criação e desenvolvimento de videojogos. É usado principalmente por equipas pequenas para criação de jogos para computadores, consolas e dispositivos móveis. A principal funcionalidade de um motor de jogo é que engloba um motor gráfico para gráficos 2D e 3D, um sistema de física, sistema de som, sistema de *script*, animação, sistema de inteligência artificial, *networking*, entre outros. O processo de desenvolvimento de jogos é, assim, acelerado, ao reusar/adaptar os módulos do motor de jogo para vários jogos, ou para lançar o jogo em várias plataformas.

Atualmente existem muitos motores de jogo disponíveis. Eles variam em muitos aspetos, desde a principal linguagem de programação, à linguagem de *scripting*, se é *cross-platform*, se é orientado a 2D/3D, a plataforma alvo e o tipo de licença.

Segundo a *Statista* [5], os cinco motores de jogo mais utilizados no Reino Unido são: *Unity 3D* (62%), *Unreal Engine* (12%), *Cocos2d* (9%), *CryEngine 3* (5%), *Marmalade* (5%).

Neste projeto foi usado o *Unity*, cujos detalhes vão ser descritos na próxima subsecção. Relativamente ao *Unreal Engine*, é um motor de jogo desenvolvido pela *Epic Games* que foi usado pela primeira vez em 1998. Desde então tem vindo a ser melhorado, estando atualmente na versão 4. Tem como linguagens principais o C++ e o *UnrealScript*. Já o *cocos2D* foi lançado em 2008 e é *open source*. Este tem várias versões, cujas diferenças entre si são as plataformas alvo e as linguagens de programação (*Python*, C++, *ObjectiveC*, C#, *Javascript*, entre outras). O *CryEngine* é o motor de jogo usado no *Far Cry*, desenvolvido pela *Crytex*. Foi tornado gratuito a 19 de agosto de 2011 e tem como linguagens principais o C++, *Lua* e o C#. Por fim, o *Marmalade* é um motor de jogo desenvolvido pela *Marmalade Technologies Limited*. Tem como linguagens de programação C/C++ e suporta um grande leque de plataformas.

O *Unity* foi o motor de jogo usado para o desenvolvimento do projeto. Inicialmente foi pensado usar-se o *Unity* ou o *Unreal Engine*. No entanto, a quando do desenvolvimento do projeto, o *Unreal Engine* não era gratuito e para além disso tinha uma curva de aprendizagem bastante elevada. O *Unity* tinha algumas vantagens comparativamente aos concorrentes: não só tinha uma curva de aprendizagem bastante baixa, como também tinha um grande suporte *online* e muitos tutoriais por onde se podia aprender. Adicionalmente, tinha a possibilidade de programar os *scripts* em C#.

IMPLEMENTAÇÃO DO PROTÓTIPO

O *Warcraft 3* foi um jogo desenvolvido pela *Blizzard Entertainment* em 2002, que recebeu a sua primeira e única expansão em 2003. É um jogo de estratégia em tempo real (com alguns elementos *Role-Playing Game*) para computador. Foi um jogo muito bem recebido pela comunidade, não só por ser o terceiro jogo da saga *Warcraft*, mas também por trazer uma inovação no modo multi jogador, a adição de criaturas

controladas pelo computador. Além disso, trouxe também um editor de mapas, no qual a própria comunidade pode criar mapas customizados e jogá-los a partir da plataforma *online*.

A partir da customização dos mapas nasceu um dos mais conhecidos mapas do *Warcraft 3*, o *Defense of the Ancients: Allstars (DOTA)*. Este mapa tem sido atualizado ao longo dos anos, tendo a sua última atualização sido lançada em 27 de Março de 2015. Foi este mapa que fez surgir dois dos maiores *Multiplayer Online Battle Arena (MOBA)* da atualidade, o *League of Legends* e mais tarde o *Dota 2*.

Foi com esta premissa que foi pensando desenvolver o protótipo baseado no *Vampirism Fire*, criando assim um *Standalone* com funcionalidades parecidas.

O *Vampirism Fire* é jogado por duas equipas. Uma equipa é composta por um máximo de oito jogadores (daqui em diante chamados de humanos) e a outra é composta por um máximo de dois jogadores (daqui em diante chamados de vampiros).

O objetivo dos humanos é angariar recursos pelo mapa, construir a sua base num dos vários pontos disponíveis, defendê-la com torres, que atacam o inimigo automaticamente, e treinar um herói, de forma a derrotar os vampiros. O objetivo dos vampiros é encontrar as bases dos humanos pelo mapa e destruí-las, ganhando mais poder por cada unidade/edifício que destroem.

Cada humano tem uma unidade principal (o trabalhador), que é o que tem a possibilidade de criar edifícios/reparar edifícios. Cada jogador só tem um trabalhador e não pode produzir mais. Para além desta unidade, existe o recoletor, que serve para apanhar os recursos (árvores neste caso). Tem ainda uma unidade especial, o herói, que durante o desenvolvimento vai ser um tanque. O vampiro apenas tem a sua unidade principal, o herói.

O trabalhador pode construir três tipos de edifícios: a refinaria, que vai servir para construir recoletores e estes irem recolher madeira; a fábrica de guerra, que vai permitir ao jogador construir os seus tanques; e a torre de defesa, que irá atacar os inimigos mal estes se aproximem dela.

Existem dois tipos de recursos neste jogo: a madeira, que serve para construir novos edifícios, melhorar os mesmos e construir unidades; e a população, que vai limitar o treino de unidades. Cada humano só pode possuir no máximo um herói. A população do jogo pode ser aumentada até um máximo de duzentos através da construção de refinarias. Ganham os humanos se matarem os vampiros ou, por outro lado, ganham os vampiros se destruírem todos os trabalhadores.

Funcionalidades Implementadas

Para que este protótipo pudesse ser testado, foi necessário implementar as seguintes funcionalidades:

1. Movimento da câmara - a câmara move-se com os movimentos do rato e quando é usado o botão de scroll, a câmara desce ou sobe, o que permite fazer o zoom in ou zoom out, dependendo da direção em que foi feito o scroll;
2. Interface Gráfica - informa o jogador do que pode ou não fazer, permitindo-o interagir com vários botões. Foram

também implementadas vários tipos de cursores, que vai transmitir ao jogador as ações que este pode tomar, um mini mapa e tooltips, que dão informação extra ao jogador;

3. Cor diferente para cada jogador;
4. Barra de vida - informa o jogador de uma forma visual a quantidade de vida das suas unidades;
5. Ações associadas aos botões do rato - selecionar e dar ações às unidades (mover, atacar, apanhar recursos);
6. Construção de edifícios e treino de unidades;
7. *SpawnPoint*;
8. Sons associados às varias ações disponíveis;
9. Condições de vitória e derrota;
10. Inteligência Artificial - torres atacam as unidades inimigas automaticamente, desde que estejam dentro do seu raio visual;
11. *Navigation Meshes* - indicar às unidades de forma inteligente para onde é que elas se podem ou não mover;
12. *Steering Behaviours* - quando se ordena o movimento de um grupo de unidades para um sítio, elas não colidem entre si, ajustando a sua posição final de acordo com a posição final das outras;
13. Sistema de grelha - indica ao jogador o sítio onde pode e não pode construir as suas unidades;
14. Fog of War - jogadores podem apenas ver uma região à volta das suas unidades.

É possível ver uma imagem do protótipo implementado na figura seguinte.

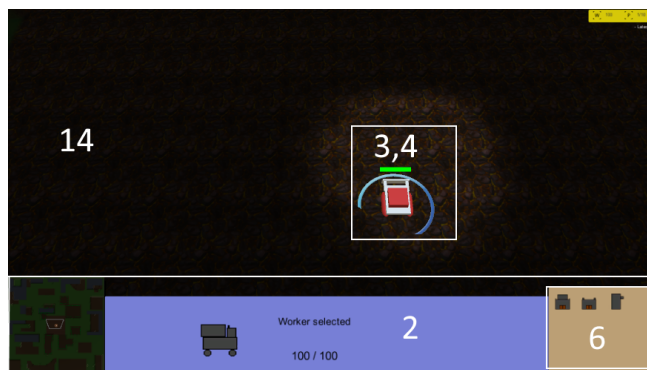


Figura 1: Protótipo implementado

Implementação do Networking

O *Networking*, apesar de ter sido a última componente a ser implementada, foi uma componente fundamental neste projeto, uma vez que os testes recaíram sobre esta componente. Para esta implementação, foi preciso criar um *player prefab*, um objeto em que vai ser instanciado sempre que um cliente se ligar ao jogo. Foi necessário criar uma nova cena com a

finalidade de fazer a ligação entre os jogadores e mandá-los para a cena de jogo.

Esta nova cena, com nome de Lobby, continha apenas um objeto vazio com duas componentes, o *Network Manager* e um *Network Manager HUD*. Foi também preciso adicionar algumas configurações ao *Network Manager*, tais como indicar que objeto seria o *player prefab* e quais seriam os prefabs que iriam ser gerados durante o jogo.

Chegando a esta parte do projeto foi preciso criar as duas versões para fazer os testes com os utilizadores. Foi feita uma cópia do projeto para que fosse possível criar as duas versões. O desenvolvimento deste projeto seguiu a ideologia *peer-to-peer*, ou seja, a autoridade dos objetos estava no lado dos jogadores. Neste caso, o servidor servia apenas para criar a sala de espera do jogo.

Para a implementação da arquitetura cliente-servidor houve necessidade de fazer mais alterações. Inicialmente teve de ser retirada a autoridade dos objetos por parte dos clientes. Esta alteração foi simples, bastou tirar o visto de “Local Player Authority” em todos os prefabs.

Após isso teve de ser feita uma alteração a todas as funções, porque qualquer ação que fosse feita tinha de passar pelo servidor (atacar, mover, recolher recurso). Assim tiveram de ser alteradas as funções que tinham a tag Command, para que toda a ação fosse comandada pelo servidor e depois replicada por todos os jogadores. Visto que as funções já estavam todas feitas, foi uma alteração rápida.

O último aspeto que teve de ser levado em consideração foi a forma como estavam a ser feitas as verificações de a quem pertencia a unidade/edifício. Até aqui eram feitas a partir da função “hasAuthority”, o que agora iria retornar sempre *false*, porque quem tinha autoridade era o servidor. A função usada passou a ser *isLocalPlayer*, para que o jogador apenas pudesse controlar as suas unidades. O instanciamento de unidades passou a ser feito por *NetworkServer.Spawn* em vez de ser feito por *NetworkServer.SpawnWithClientAuthority*.

Faltava apenas criar algo para que os jogadores pudessem escolher a sua equipa. Foi usado como base o projeto *Lobby Manager* do *Unity*. As alterações feitas a este *LobbyManager* ocorreram ao nível da implementação de novas funcionalidades sobre ele, já que ele trata de todas as ligações entre os jogadores.

Este lobby, que apenas fazia com que vários jogadores se juntassem a uma sala de espera antes de começar o jogo, sofreu várias alterações, em que foi implementado as duas equipas, correção das cores de fundo dos painéis, implementação de scrollbar se houver muitos jogadores, apenas o hoster pode começar a partida, e só se todos os jogadores estiverem prontos e os jogadores podem cancelar o seu estado de pronto.

PROTOCOLO DOS TESTES

Foi criado um teste para verificar qual das duas arquiteturas implementadas funciona melhor no protótipo desenvolvido.

Quarenta utilizadores realizaram os testes ao protótipo desenvolvido, tendo avaliado as duas versões, cliente/servidor e

peer-to-peer. Visto que a latência é um fator que influencia este tipo de testes, estes não foram realizados presencialmente, mas sim com cada uma das pessoas em sua casa, usando o seu computador pessoal. Assim, os jogadores não se ligaram à sala de espera do jogo através de uma ligação local, mas sim através de uma ligação a partir da Internet, ou seja, uma ligação real. Durante os testes, toda a conversação foi realizada mediante uma ligação *Skype* e a partida foi gravada para posterior análise.

Os testes foram aplicados a grupos de quatro pessoas, duas por equipa. Cada teste foi dividido em dois pequenos sub-testes, que serviram para avaliar os dois tipos de arquitetura. No teste da arquitetura cliente/servidor existiu um servidor dedicado para testar o protótipo, enquanto que no teste da arquitetura *peer-to-peer*, como a comunicação é feita entre os jogadores, apenas houve necessidade de um servidor para fazer a sala de espera do jogo.

Foi construído um formulário usando o *Google Forms*, que continha informação importante para os participantes que iriam realizar o teste. No início do formulário fez-se uma introdução do teste aos participantes, enunciando o seguinte:

- Foi explicado o que ia acontecer durante o teste e o motivo de ele estar a ser realizado;
- Foi referido que o teste ia testar o protótipo desenvolvido e não o participante;
- Foi dito que se o participante tivesse dificuldades em usar ou compreender o que era pedido, deveria referi-lo, visto que possivelmente outros participantes iriam sentir a mesma dificuldade;
- Foi dito ao participante que, se quisesse, poderia desistir do teste a qualquer momento e que o teste duraria cerca de trinta minutos;
- Foi enunciado que todos os dados recolhidos seriam usados apenas para a pesquisa, salvaguardando-se o anonimato dos participantes, e que em caso de publicação dos dados, estes seriam compilados com a informação de todos os outros participantes.

De seguida, foi solicitado aos participantes o preenchimento de um pequeno inquérito com alguma informação, tal como o género, a faixa etária, se costuma jogar e, se sim, qual o tipo de jogos que joga, quantas horas por semana e se os jogos que joga são *online* ou não.

Após esta informação foi mostrada uma lista de tarefas que os participantes deveriam completar por ordem sequencial, sem saltar nenhuma. Foram também informados que caso tivessem alguma dificuldade em completar uma das tarefas, deveriam apresentá-las no final do teste. Foi também questionado se tinham alguma dúvida antes de se dar início ao teste.

Assim que o participante acabava todas as tarefas deveria aguardar que todos os participantes o tivessem feito de modo a poder mudar de jogo (para a outra arquitetura), fazendo a mesma lista de tarefas. Após ter completado todas as tarefas usando as duas arquiteturas, deveria passar para a última

página do formulário, onde lhe era perguntado se tinha compreendido todas as tarefas, os objetivos, se conseguiu perceber como interagir com o protótipo e questões para comparar os dois tipos de arquitetura.

Resultados

Dos quarenta participantes, três eram raparigas, duas delas com uma idade superior a 27 anos e uma entre os 18 e os 20 anos. Os restantes participantes eram rapazes, treze entre os 18 e os 20 anos, catorze entre os 21 e os 23, oito entre os 24 e os 26 e dois com idade superior a 27 anos.

Entre os participantes, cinco deles não costumavam jogar no dia a dia. Dos restantes, dezasseis costumavam jogar jogos de estratégia, enquanto que dezanove estavam igualmente divididos entre *MOBA's* e *MMORPG*.

Por semana, três dos participantes jogavam menos de três horas, sete jogavam entre três a seis horas, dez jogavam entre seis e nove horas, cinco jogavam entre nove e doze horas e dez jogavam mais de doze horas. Dos trinta e cinco participantes que jogavam, dois não costumavam jogar jogos *online*.

Relativamente às perguntas sobre o teste, dos quarenta participantes, trinta e nove perceberam o objetivo de protótipo e um não. No entanto, todos eles perceberam como interagir com o protótipo, inclusive os cinco que não costumam jogar. Relativamente à pergunta sobre se tinham percebido as tarefas que foram pedidas, todos os participantes compreenderam à exceção de um, que referiu que não tinha percebido como atacar as unidades adversárias.

De modo geral, os problemas indicados pelos participantes puderam dividir-se em três categorias:

- *Fog of war* - alguns participantes sentiram dificuldades em encontrar as suas unidades, visto que a área de abertura no *fog of war* estava demasiado pequena. Este problema intensificava-se nas plataformas;
- Criação de unidades nas bordas das plataformas - se um participante construísse um edifício na borda de uma plataforma e treinasse uma unidade, existia a possibilidade de ela ir para a parte de baixo da plataforma;
- Falhas no movimento das unidades e no ataque - algumas unidades não respondiam corretamente à ordem de ataque e não evitavam as colisões da melhor maneira.

Um participante não conseguiu jogar o modo cliente/servidor. Embora não exista total certeza sobre qual a origem do problema, o participante estava ligado à rede *MEO-WIFI*. Houve também três participantes que perderam a ligação com o servidor (modo cliente-servidor). Estes três participantes estavam a partilhar a ligação à Internet com um quarto que não perdeu a ligação.

Em relação à pergunta se notaram diferença entre os dois testes efetuados, dezoito participantes responderam que não, enquanto vinte e dois afirmaram que sim. Relativamente às diferenças encontradas, as respostas foram unânimes, os jogadores referiram que sentiram menos lentidão na arquitetura *peer-to-peer* comparativamente à arquitetura cliente/servidor,

assim como uma maior fluidez do protótipo (*frame rate* constante) e maior rapidez de resposta das unidades. Já na arquitetura cliente/servidor as unidades demoravam algum tempo a realizar as ordens dadas, quer fossem de movimento ou de ataque.

Relativamente à fluidez do protótipo, trinta e quatro participantes disseram que a arquitetura *peer-to-peer* era fluída, enquanto que os outros seis referiram sentir alguma lentidão. Para além disso, dezassete deles afirmaram que a arquitetura cliente/servidor também era fluída, enquanto que os restantes vinte e dois disseram que era lenta. Um participante não respondeu a esta pergunta pois não conseguiu testar este modo.

Já na última pergunta, a qual questionava se os participantes tinham notado alguma demora por parte das unidades em reagir quando era dada alguma ordem, vinte e seis referiram não terem sentido nenhuma lentidão, enquanto que catorze disseram que sim, relativamente à arquitetura *peer-to-peer*. Já na arquitetura cliente/servidor, dezasseis disseram que não sentiram nenhuma lentidão, enquanto que vinte e dois afirmaram que sim. Um participante não respondeu a esta pergunta pois não conseguiu testar este modo.

PRINCIPAIS CONCLUSÕES E TRABALHO FUTURO

Com base nos resultados obtidos através dos testes efetuados com quarenta participantes, pode-se concluir que a arquitetura *peer-to-peer* no *Unity* funcionou melhor para a maioria dos participantes, em que apenas uma pequena minoria sentiu alguma lentidão neste modo. Esta lentidão poderá não ser causada pela arquitetura de rede, mas pela ligação à Internet do participante ou por causa do participante estar a usar um computador com algumas limitações. Este resultado difere dos resultados obtidos na arquitetura cliente-servidor, onde a maioria dos participantes disseram que este modo era lento.

De forma a tentar obter resultados mais conclusivos relativamente a estas duas arquiteturas, seria necessário realizar os testes em ambiente controlado, ou seja, todos os participantes deveriam usar computadores com especificações iguais e usarem uma ligação à Internet igual, estando todos no mesmo local. Poderia também ser usada uma ligação dedicada de *1gb* por segundo de *upload* no servidor (foi usada uma ligação dedicada de *100mb* por segundo de *upload* neste teste).

Poderiam também ser feitos testes automáticos, usando as *test tools* do *Unity*, de forma a simular uma partida com vários jogadores e conseguir perceber qual das duas arquiteturas funciona melhor, sem ser necessário ter pessoas a jogar presencialmente. Poderiam também ser feitos testes usando uma arquitetura híbrida, apesar de o *Unity* não fazer nenhuma referência se é possível a sua implementação.

O facto de o *Unity* apenas ter documentação e tutoriais de implementações usando a arquitetura *peer-to-peer* pode indicar que esta seja a melhor arquitetura a ser usada na implementação de jogos multi jogador ou que deverá ser a arquitetura privilegiada pelos programadores.

AGRADECIMENTOS

O autor filiado ao Instituto de Telecomunicações agradece o suporte dado através do programa FCT projeto UID/EEA/50008/2013.

REFERENCES

1. M. Badar and T. Nikhil. 2013. Development Guidelines for Mobile Multiplayer Games. (2013), 26.
2. R. Humphrey, A. Allan, and G Di Fatta. 2012. Using Spatial Locality and Replication to Increase P2P Network Performance in MMO Games. (2012), 6.
3. M. Powers. 2006. Mobile Multiplayer Gaming, Part 1: Real-Time Constraints. (2006).
4. N. Sheldon, Girard E., S. Bord, M. Claypool, and E. Agu. 2003. The Effect of Latency on User Performance in Warcraft III. (2003), 12.
5. Statista. 2014. Leading game engines used by video game developers in the United Kingdom (UK) 2014. (2014). <http://www.statista.com/statistics/321059/game-engines-used-by-video-game-developers-uk/>
6. A. I. Wang, M. Jarrett, and E. Sorteberg. 2009. Experiences from implementing a mobile multiplayer real-time game for wireless networks with high latency. (2009), 15.

Análise heurística de jogos digitais multijogador em rede: O caso do “Curve Fever 2”

Cláudio Duarte
Universidade de Aveiro
claudioduarte@ua.pt

Ana Isabel Veloso
Universidade de Aveiro
Digimedia (CIC.Digital)
aiv@ua.pt

RESUMO

A avaliação de jogos digitais é uma fase fundamental no ciclo de produção dos jogos digitais. A avaliação heurística é uma das técnicas para efetuar essa avaliação. O objetivo deste artigo é apresentar uma avaliação heurística ao jogo multijogador em rede “Curve Fever”. A avaliação foi realizada por 5 avaliadores com base em 10 heurísticas de usabilidade construídas para a avaliação de jogos multijogador em rede. Os resultados revelaram que o jogo pode ser melhorado nalguns aspetos, nomeadamente, a existência de um espaço para os novos jogadores treinarem, a redução dos atrasos de *feedback* das ações do jogador e a melhoria dos avatares.

Palavras-Chave

Avaliação heurística, Jogo digital, Jogo multijogador em rede, Usabilidade

INTRODUÇÃO

O fracasso dos produtos digitais por vezes advém do facto de não ter sido efetuado um plano de testes e uma avaliação adequada. Os jogos digitais não são exceção.

Segundo [1] e [2], podem ser efetuados diferentes tipos de teste com diferentes objetivos, nomeadamente, os testes ao equilíbrio da jogabilidade para perceber se existe equilíbrio justo entre a jogabilidade dos jogadores e dos elementos com inteligência artificial; os testes de compatibilidade, muito importantes devido aos diferentes sistemas operativos que os computadores possuem; os testes de conformidade que lidam com a certificação dos produtos; os testes de localização devido à necessidade de localização do jogo para os diferentes contextos social e cultural; os testes relacionados com o “*playtesting*” para garantir a diversão do jogo; e os testes de usabilidade que pretendem garantir que é intuitivo interagir com os elementos do jogo (personagens, interfaces, inventário, ou dirigir um veículo, entre outros). Os métodos de avaliação de usabilidade são vastos, tais como, “*cognitive walkthroughs*”, entrevistas, observações, questionários, mas um dos mais comuns é a avaliação heurística.

O objetivo deste artigo é a apresentação e análise da avaliação heurística do jogo digital “Curve Fever 2”. Apresenta-se uma contextualização breve dos conceitos sobre os jogos digitais, as heurísticas que existem para avaliar jogos digitais e as versões existentes do jogo “Curve

Fever”. No método apresenta-se a avaliação heurística realizada e finaliza-se com os resultados e conclusões.

JOGOS DIGITAIS

Segundo [3], um jogo é “um sistema no qual jogadores envolvem-se num conflito artificial definido por regras que resulta num resultado quantificável”. Esta definição resulta de um levantamento geral de elementos essenciais para atingir uma definição de jogo, segundo as diversas perspetivas de vários autores, sendo que a presença de regras que limitam o jogador e a orientação para um determinado objetivo, são elementos cuja presença em jogos digitais aparenta serem fundamentais. Em contrapartida, a forma de arte de Costikyan [4], a ineficiência devido às regras [5], a incerteza [6] e a promoção de grupos sociais especiais, aliada à referida atividade “não-séria” [7] expõem-se como sendo os elementos menos recorridos para uma visão geral do conceito de jogo. Três dos autores compilados [6], [8] e [3] indicam seis elementos para construir o conceito de jogo que se destacam: não associar a benefícios materiais; ser um ato voluntário; associar a necessidade de tomar decisões; e a artificialidade são apenas alguns desses elementos.

HEURÍSTICAS PARA AVALIAR JOGOS DIGITAIS

O termo “*heuristic*” (heurística) [9] significa “*shortcut*” (atalho), e é utilizado com diferentes significados em diferentes áreas de estudo. Na área das ciências da computação associa-se ao termo o conceito de usabilidade para identificarmos problemas associados à utilização de *software*. A avaliação heurística é realizada através de conjunto de orientações pré-definidas que podem influenciar o desenvolvimento de um projeto [10]. De acordo com Nielsen [11], são chamadas “heurísticas” porque são regras gerais e não orientações de usabilidade específicas. Este método de avaliação pode ser considerado flexível em comparação com outros métodos de avaliação na medida em que se identificam muitos mais problemas a um custo muito mais reduzido [12].

Na literatura existe uma corrente de autores que reconhece a importância de heurísticas para a conceção de jogos digitais [13], [14]. Segundo Federoff [13] a avaliação de jogos é uma prática relativamente recente e ainda pouco desenvolvida na indústria dos jogos. A autora destaca que já decorreram mais de vinte anos desde que foi desenvolvido

o primeiro conjunto de heurísticas específicas exclusivas aos jogos baseadas nos seguintes elementos: Desafios, Curiosidade e Fantasia [15], e a sua presença ainda é mínima.

De um modo geral, a avaliação de usabilidade nos jogos não é um método muito recorrido e, segundo indica Federoff [13], descobriu-o no decorrer da sua investigação uma vez que os participantes tiveram dificuldade em identificar o próprio termo de usabilidade. O estudo desenvolvido consistiu numa extensa revisão da literatura sobre heurísticas para usabilidade de jogos e num estudo de campo realizado numa empresa de videojogos. Deste estudo resultou um conjunto de 40 heurísticas de análise em três categorias. A autora destaca que segundo Chuck Clanton é a forma de englobar a maioria dos problemas relacionados com usabilidade em jogos em três campos: interface do jogo (*game interface*), mecânica de jogo (*game mechanics*) e jogabilidade (*game play*). A interface do jogo diz respeito ao meio utilizado pelo utilizador para interagir; a mecânica é “a física do jogo” que agrega a programação e a animação; e a jogabilidade corresponde à meta principal a atingir pelo jogador.

Com a evolução dos jogos digitais foram surgindo recentemente diversos estudos que construíram heurísticas para o design e avaliação de jogos digitais em diferentes contextos e dispositivos, tais como, dispositivos móveis [16], jogos multijogador em dispositivos móveis [17] jogos nas redes sociais [18], entre outros.

As heurísticas desenvolvidas por Pinelle *et al.* [14] foram construídas fruto dos problemas descritos nos relatórios sobre análise de 382 jogos para computador em rede, cobrindo seis géneros principais. Estas 10 novas heurísticas servem para orientar o *design* e a avaliação de jogos multijogador em rede.

O JOGO “CURVE FEVER”

O “Curve Fever”, criado a 4 de abril de 2010 em Breda, Holanda, apresenta-se como sendo um jogo digital *multiplayer* em rede [19]. As principais personagens são “cobras” e, como objetivo, o jogador deve cortar outros jogadores com a sua “cobra”, tentando manter-se vivo o maior tempo possível. Ao longo do jogo, os jogadores vão usufruindo de *power-ups* que aparecem na área de jogo e que auxiliam na evolução durante rondas. Podem afetar, pela positiva ou negativa, tanto o jogador em si como os restantes na arena. Ganha o jogador que obtiver primeiro um total de 35 pontos ou, em caso de empate, aquele que alcançar um resultado superior de 2 pontos.

História

Inicialmente concebido por mera diversão pelo programador Geert van den Burg, segundo a *press release* do jogo [19], o “Curve Fever” não teve o sucesso previsto após o seu lançamento online. No entanto, após a sua má receção, o colega de casa de Geert apresentou o jogo aos

seus amigos, que o jogaram todos os dias durante um mês; foi a inspiração necessária para que Geert continuasse o seu desenvolvimento. Após uma semana intensa de programação, é lançada a primeira versão oficial numa página *web* dedicada – o “Curve Fever 1” – já que nenhum *website* de jogos se interessou em o alojar nos seus servidores.

Evoluindo lentamente, mas de forma progressiva, o “Curve Fever 1” foi ganhando popularidade juntos dos seus utilizadores. No entanto, este avanço exigiu que o jogo fosse reformulado, de modo a chegar ainda a mais potenciais jogadores. A aposta incidiu na criação de uma versão *multiplayer*¹ do jogo, possibilitando a qualquer pessoa jogar sozinha no seu próprio teclado sem que existisse a necessidade de o partilhar com mais utilizadores. Com esta mudança, surgiu a segunda versão do jogo – “Curve Fever 2” – tornando-o viral. Recebe, atualmente, cerca de 50.000 jogadores por dia, sendo que a maioria aproveita os intervalos do período escolar para jogar [19].

Paralelamente, encontra-se em desenvolvimento o “Curve Fever 3”, uma nova versão mais atualizada que, segundo Geert, “trará um novo e fresco Curve Fever” com “novos e interessantes elementos de jogabilidade” [20]. O seu lançamento está previsto para o terceiro quadrimestre de 2016.

A primeira versão: “Curve Fever 1”

O “Curve Fever 1”, gratuito e atualmente ainda disponível ao público apresentava, apesar das suas limitações aquando do seu lançamento, uma boa jogabilidade; não era, contudo, a mais prática, sendo que para jogar com mais jogadores em simultâneo (seis ao todo) era necessário partilhar, obrigatoriamente, o mesmo teclado.

No que diz respeito ao *design* de interface, este apresenta-se como pouco apelativo devido à escolha da organização e hierarquização de conteúdo, não permitindo que este “respire”. A informação referente aos jogadores encontra-se muito próxima da área de jogo. Para além disso, o facto de serem utilizadas mais do que uma fonte tipográfica que não se complementam, torna a interface ainda mais inconsistente. Em contrapartida, algumas funcionalidades positivas a referir são a possibilidade de colocar o jogo em pausa, assim como a existência de *power-ups*², apesar de limitados.

¹ Multijogador

² Expressão utilizada para referir um item que aumenta uma determinada característica de algum utilizador (ex: velocidade/tamanho).

A segunda versão: “Curve Fever 2”

O “Curve Fever 2”, já numa versão de *fremium*³, evidencia uma intervenção mais cuidada, apostando na evolução do jogo e no *design*. Foi adicionado ao jogo a funcionalidade de *multiplayer* permitindo que este chegasse a mais utilizadores e ainda o facto de estar *online* converteu-o num jogo em rede. Tornou-se possível iniciar uma competição com utilizadores presentes nos quatro cantos do mundo, alargando ainda mais o leque de membros que passam a usufruir do jogo.

A nova versão trouxe, adicionalmente, melhoramentos a vários níveis: personalização [limitada] de ícones, cores das cobras e *nickname*⁴ do jogador, criação de salas de jogo com mais jogadores (até oito), *chat* e lista de amigos, *matchmaking*⁵, clãs, entre outros. Ao nível do *design* de interface, este apresenta-se mais limpo, organizado e hierarquizado de uma forma mais adequada, tornando a informação presente na área de jogo mais legível. Os recursos tipográficos foram, igualmente, alvo de normalização.

Esta segunda versão do “Curve Fever” será a versão alvo da avaliação heurística deste artigo. A competitividade saudável, a interação e criação de novas amizades, a evolução e posicionamento no jogo, a possibilidade de convidar amigos próximos para jogar em conjunto são alguns motivos da escolha.

A terceira versão: “Curve Fever 3”

O “Curve Fever 3” ainda se encontra em desenvolvimento; o seu lançamento estava previsto para março de 2016, tendo, contudo, esse prazo sido adiado para os finais do ano civil. A nova versão do jogo pretende vir a melhorar a interação social e a competitividade entre os utilizadores, evitar a sensação de repetibilidade no jogo e facilitar o acesso a conteúdo, nomeadamente *power-ups*, que se encontram atualmente apenas ao alcance dos utilizadores *premium*⁶ [20].

MÉTODO DE AVALIAÇÃO

As heurísticas escolhidas [14] destacam-se pelo facto de se adequarem ao género de jogo em avaliação – um jogo *online* que estimula a participação de múltiplos jogadores que interagem socialmente por meio de uma rede para fins de entretenimento. Mais, por ser um método de avaliação

considerado mais barato [12], apresenta-se como ideal para as limitações deste estudo.

Segundo Nielsen [11] uma avaliação heurística pode ser efetuada por um ou vários avaliadores. Um avaliador tem a desvantagem de identificar apenas nas 35% dos problemas, mas entre 3 e 5 avaliadores, dependendo da extensão da avaliação, podem ser detetados até 85% dos problemas.

De modo a avaliar o jogo reuniu-se 5 avaliadores que recorreram às heurísticas de usabilidade para jogos multijogador em rede [14], apresentadas nas Tabelas 1 e 2, de modo a avaliar uma versão específica do “Curve Fever 2”. O objetivo da avaliação era identificar a qualidade do jogo e o emprego da usabilidade no mesmo de modo a compreender se este deverá ser considerado, ou não, como um bom exemplo de divertimento digital.

	Heurística
H1	Gestão simples de sessões
H2	Matchmaking flexível
H3	Ferramentas de comunicação adequadas
H4	Apoio à coordenação
H5	Consciência de Informação
H6	Avatares identificáveis
H7	Treino para iniciantes
H8	Interação Social
H9	Redução de atrasos relacionados com o jogo
H10	Gestão de mau comportamento

Tabela 1. Heurísticas de Jogos em Rede

	Descrição
H1	Fornece a gestão de sessões que permite aos jogadores criarem e iniciar novos jogos, permitindo que encontrem e se juntem a jogos apropriados
H2	Fornece recursos de <i>matchmaking</i> para ajudar os utilizadores a encontrarem jogadores com interesses semelhantes
H3	Fornecer recursos de comunicação que podem acomodar as exigências do jogo
H4	Fornecer recursos que permitem aos jogadores coordenarem as suas ações durante o jogo
H5	Fornecer informações significativas sobre os jogadores

Tabela 2. Descrição das Heurísticas de Jogos em Rede

³ Oferecido gratuitamente, mas com recursos adicionais pagos.

⁴ Nome de utilizador.

⁵ Processo de juntar duas ou mais pessoas que partilham interesses em comum.

⁶ Conteúdo acessível mediante pagamento.

	Descrição
H6	Usar avatares visíveis e distintos que têm mapeamentos de informação intuitivas
H7	Proporcionar oportunidades de treino onde os novos jogadores não estão sujeitos a pressões de jogadores mais experientes
H8	Fornecer suporte para interações sociais
H9	Minimizar os atrasos de interação, reduzindo dependências temporais entre os jogadores
H10	Fornecer soluções técnicas e sociais para a gestão de comportamento desagradável

Tabela 2. Descrição das Heurísticas de Jogos em Rede (cont.)

Pretende-se que, com esta avaliação heurística, se alcance as diferenças e/ou semelhanças entre as perspetivas distintas.

Metodologia aplicada

Os avaliadores têm características diferentes, dois deles são jogadores frequentes do “Curve Fever” e outros três jogadores não estão familiarizados com o jogo. Todos os avaliadores demonstram ser participantes assíduos de jogos em rede conhecedores deste género.

As heurísticas escolhidas foram dadas a conhecer aos sujeitos dois dias *a priori* da avaliação. É de referir que estas heurísticas podem ser utilizadas por qualquer multijogador, desde que sejam dadas a conhecer num período de tempo significativo antes de uma avaliação de modo a permitir que o avaliador se familiarize com as mesmas.

Aos participantes do estudo foram entregues, em formato digital, as heurísticas em causa de um modo detalhado, numa tabela elaborada pelos seus autores [14]. Foi facultado um documento *Word*, previamente concebido, aos participantes, permitindo que registassem a sua avaliação individual de cada heurística.

Foram realizadas duas sessões de avaliação. Cada sessão tinha a duração de trinta minutos. Na primeira, foi solicitado aos avaliadores que se focassem no jogo em questão, analisando inicialmente a interface de menu do “Curve Fever 2” de modo a fazer um levantamento geral do seu aspeto, tanto a nível de *design* como de funcionalidades disponíveis. De seguida, iniciavam um jogo, quer por *matchmaking* ou *custom*, onde poderiam escolher as salas em que gostariam de participar. Foi, simultaneamente, lançado o desafio de criarem uma sala sua, avaliando o processo associado.

O objetivo da segunda sessão era jogarem uma a duas vezes o jogo, completando todas as rondas até à atribuição da pontuação final enquanto procediam, em simultâneo, à análise da interface da área de jogo e as funcionalidades que apresentava.

Durante todo este procedimento, caso algum aspeto positivo ou negativo com base em alguma heurística surgisse, solicitou-se que registassem no documento fornecido e que descrevessem brevemente o problema ou a mais-valia. O documento redigido pelos avaliadores serviria o propósito de apresentação das heurísticas associadas ao “Curve Fever 2”, focando na presença e influência das mesmas no jogo.

RESULTADOS

A análise dos documentos redigidos pelos avaliadores permitiu ver se partilhavam, ou não, uma visão comum sobre a avaliação do “Curve Fever 2”. Destaca-se assim de seguida o impacto de cada heurística no jogo, assim como a opinião geral fornecida pelos avaliadores.

À semelhança de Pinelle, Wong, Stach e Gutwin [14], mesmo através de uma simples investigação, “as avaliações de usabilidade em jogos multijogador podem ter um grande impacto no sucesso de um jogo em rede”. O “Curve Fever 2” apresentou vários aspetos positivos e negativos avaliados num total de 10 heurísticas. Fruto da avaliação feita pelos sujeitos avaliadores houve, em pelo menos metade das heurísticas, concordância entre os sujeitos, apresentando estas mais-valias; no que diz respeito às restantes heurísticas, partilham também problemas.

Aplicação da avaliação heurística

Gestão simples de sessões (H1). Começando logo pela heurística H1, é possível jogar o “Curve Fever 2” sendo visitante ou utilizador registado, havendo flexibilidade de escolha. É dada a possibilidade de entrar em salas já criadas ou criar a sua própria com definições apropriadas permitindo, por exemplo, limitar os utilizadores que acedem à sala protegendo-a com uma palavra-passe. Nestas salas é possível consultar os utilizadores que pretendem jogar, assim como o seu *ranking*, a cor atribuída à sua cobra, entre outros. Assim, ao nível do *ranking*, o utilizador pode consultar o de outros jogadores e perceber se a sala que escolheu será a mais adequada para o nível de *expertise*⁷. A qualquer momento, no decorrer do jogo, é possível entrar, abandonar e regressar a uma sala. Na perspetiva dos restantes avaliadores, esta heurística apresenta as características positivas mencionadas.

Matchmaking flexível (H2). Decorrente da avaliação, o “Curve Fever 2” permite ao utilizador adicionar amigos à sua *buddy list*⁸, o que mais tarde poderá vir a ser benéfico, no sentido em que poderão ser lançados jogos com estes amigos que partilham interesses em comum. Para além disso, o próprio jogo integra a funcionalidade de *matchmaking* automático, onde o utilizador é conectado com outros cinco jogadores e uma sala é atribuída de

⁷ Experiência.

⁸ Lista de amigos.

imediatamente onde de seguida se inicia o jogo. No entanto, existe outra funcionalidade - jogos *custom* - como já mencionado na heurística H1, onde o utilizador poderá criar uma sala e seleccionar os jogadores que pretende que a integrem. Dois dos avaliadores (3 e 4) partilham a mesma opinião no que diz respeito ao tempo necessário para completar o *matchmaking*, classificando-o como demorado, o que poderá levar à desistência desta opção por parte de alguns utilizadores, optando pelo jogo *custom*.

Ferramentas de comunicação adequadas (H3). O “Curve Fever 2” dispõe de formas de *feedback*, tanto ao nível visual como sonoro, permitindo que os utilizadores estejam constantemente informados dos acontecimentos que estão e vão decorrendo ao longo de todo o jogo. A indicação visual da captação de *power-ups* e da pontuação de cada jogador ou a indicação sonora dada ao jogador quando este não se encontra focado na área de jogo são alguns exemplos. Todos os avaliadores concordam com esta observação, apontando esta heurística como bastante positiva, visto que apela à concentração do utilizador para as exigências do jogo.

Apoio à coordenação (H4). Em relação a esta heurística, a avaliação verificou claramente que existe coordenação dentro do jogo que é feita, exclusivamente, com recurso ao teclado. Quando desloca a sua “cobra” na área de jogo, o jogador tem de planear para que ponto se deve mudar; à medida que a cobra percorre a área de jogo, vai deixando um rasto para trás, ocupando espaço que não poderá ser ocupado por outras “cobras”, obrigando a repensar a estratégia de desvio de forma constante. Os avaliadores 2, 3, 4 e 5 partilham da mesma opinião, mas o avaliador 2 salienta que a coordenação no jogo será mais visível quando jogado em modo de equipa enquanto o avaliador 4 indica que a associação entre o utilizador e a sua “cobra” não é, logo à partida, perceptível, induzindo o utilizador em erro. O jogo potencia igualmente a coordenação do utilizador pelo facto de distribuir diferentes *power-ups* ao longo do jogo que podem tanto beneficiar como prejudicar o utilizador; no entanto, o avaliador 3 indica, na sua avaliação que apenas depois de apanhar alguns destes *power-ups* é que percebeu a sua utilidade e a influência que tinham no jogo e no utilizador.

Consciência de Informação (H5). No que diz respeito a esta heurística, todos os sujeitos apontam aspetos positivos. Antes e durante o jogo é possível localizar e consultar informações provenientes de outros utilizadores. A plataforma disponibiliza uma funcionalidade de pesquisa de jogadores permitindo que qualquer pessoa encontre um jogador, desde que conheça o seu *username*. Com esta funcionalidade é possível a alguém juntar-se a uma sala de jogo e jogar com a pessoa que procurou, algo bastante útil para amigos que queiram jogar juntos ou em equipa. As informações acessíveis de cada utilizador incluem as conquistas ganhas (leia-se *achievements*), pontos de equipa/FFA/1versus1, número de vezes que já jogou e a

percentagem de jogos ganhos. Já dentro da área de jogo propriamente dito, existe um *scoreboard* que possui informação síncrona, ou seja, o que acontece ao longo do jogo e a posição do jogador, como assíncrona (com o *ranking* e a informação dos jogadores).

Avatares identificáveis (H6). Esta heurística tem alguma presença no “Curve Fever 2” mas não é muito significativa na medida em que o jogo, mesmo assim, apresenta aspetos negativos. Face às avaliações, o avatar não é um direito adquirido logo à partida. Cada jogador poderá a vir ter um avatar, no entanto, apenas mediante um pagamento com a moeda do jogo. Para além desse aspeto, não existe personalização do avatar, estando o utilizador limitado à oferta disponibilizada: branco como cor principal, tamanho pequeno por defeito, oferta de ícones limitada. A presença desse avatar durante um jogo, por exemplo, é reduzida, passando a penas a estar visível no *scoreboard* do jogo ao invés de estar presente na área de jogo. Aí, é a cor da cobra do jogador que o identifica. Portanto, a presença do avatar no jogo não é bem conseguida. O “Curve Fever 2” beneficiaria em apresentar um *upgrade* ao nível dos avatares, permitindo uma maior personalização e presença de modo a que os utilizadores se identifiquem mais com eles.

Treino para iniciantes (H7). O “Curve Fever 2” não apresenta qualquer funcionalidade dedicada ao treino do jogo para jogadores principiantes. Todos os sujeitos avaliadores indicaram esta falha como algo grave, sendo que os novos utilizadores não se sentem à vontade para explorar o jogo, estando obrigados desde logo a enfrentar jogadores mais avançados. Este fenómeno poderá levar a que muitos potenciais jogadores desistam completamente de jogar. A possibilidade de jogar como visitante aproxima-se a este treino para iniciantes, no entanto, acredita-se que o utilizador registado deveria ter a possibilidade de começar, desde logo, a ganhar posicionamento no jogo, mesmo na fase de treino. Jogos como o Agar.io permitem precisamente isso, disponibilizando uma sala dedicada ao treino. É fundamental existir uma área dedicada aos novos utilizadores para que estes se sintam livres de explorar o “Curve Fever 2” sem se sentirem pressionados. Assim, o jogo ganhará rapidamente mais adeptos.

Interação Social (H8). A presença constante de um meio de comunicação entre jogadores – o *chat* – é bastante benéfico para este jogo que conta com utilizadores de vários cantos do mundo. A interação proporcionada fomenta a partilha e criação de novas amizades dentro do jogo. O *chat* está presente fora do jogo (entenda-se fora das salas), antes de iniciar um jogo (enquanto aguardamos na sala), durante o jogo e no final do mesmo. Isto permite que os utilizadores estejam em constante diálogo quebrando barreiras entre os jogadores de modo a incentivar uma participação ativa.

Redução de atrasos (H9). No que diz respeito aos atrasos ou “lag”, todos os sujeitos avaliadores partilham da mesma

opinião indicando que o jogo, por si só apresenta, com grande frequência, muitos atrasos. A necessidade de rede por parte do jogo é bastante elevada devido a todos os recursos com que o jogo operacionaliza. Acredita-se que este fenómeno leva a que muitos potenciais jogadores acabem por desistir de jogo pelo simples facto de a sua rede de Internet não corresponder às exigências do jogo. No entanto, referenciando novamente o Agar.io, o jogo aposta na redução do “lag” através da redução da qualidade dos gráficos do jogo, permitindo que mais rede seja dedicada à jogabilidade. Para o “Curve Fever 2”, acredita-se que seria uma mais-valia adotar um procedimento semelhante. Por outro lado, a aposta em mais e melhores servidores poderá ajudar igualmente com este problema.

Gestão de mau comportamento (H10). Uma das funcionalidades mais admiradas pelos sujeitos avaliadores no “Curve Fever 2” é a presença de um tribunal onde é possível denunciar maus comportamentos por parte de outros jogadores. Caso um determinado jogador seja reportado por vários utilizadores sobre o mesmo ou outros assuntos, será aberto um processo onde a situação será analisada por voluntários. Os casos analisados são maioritariamente oriundos de conversas inapropriadas no chat do jogo.

Resumidamente, a avaliação feita pelos sujeitos avaliadores, com base nas heurísticas apresentadas, pode-se traduzir em pontos positivos e negativos demonstrados na Tabela 3 com base nas mais-valias e falhas que o “Curve Fever 2” apresenta. Sendo assim, é possível detetar rapidamente em que heurísticas o jogo apresenta uma boa ou má aposta.

Logo à partida verifica-se a existência de uma diferença significativa entre o primeiro conjunto (H1 a H5) e o segundo conjunto (H6 a H10) de heurísticas. O primeiro conjunto apresenta maioritariamente aspetos positivos, com algumas exceções devido à experiência dos avaliadores. O segundo conjunto direciona-se para aspetos negativos sendo que está relacionado com problemas do jogo em si que merecem ser melhorados.

CONCLUSÕES

A usabilidade nos jogos digitais ainda tem um longo percurso a percorrer. Apesar de, neste artigo, o foco principal se basear nas *Usability Heuristics for Networked Multiplayer Games*, certamente haverá mais e melhores heurísticas para avaliar jogos digitais em rede. Como referido anteriormente, estas foram criadas por se sentir a falta de um guia de avaliação para o género de jogo como o “Curve Fever 2”. Tal como refere Federoff [13], uma linguagem comum terá de ser desenvolvida para que os *developers* e os profissionais dedicados à HCI – *Human-Computer Interaction* – se sintam cómodos a utilizar.

Em relação ao jogo propriamente dito, é evidente, através da avaliação heurística realizada, que existe uma concordância bastante significativa e positiva das

heurísticas com o “Curve Fever 2”. Através da avaliação verificou-se que existem pontos positivos, assim como pontos negativos, merecedores de melhorias, nomeadamente a existência de um espaço para os novos jogadores treinarem, a redução dos atrasos e a melhoria dos avatares. Ainda assim, o “Curve Fever” não deixa de ser um bom jogo, independentemente da versão. É intuitivo, interessante, estimulante e viciante.

Com o surgimento da nova versão do jogo – “Curve Fever 3” – os *developers* pretendem implementar melhorias importantes tanto ao nível do *design* de interface como da jogabilidade de modo a captar mais utilizadores para o jogo e renovar a confiança e interesses dos já registados. Esperamos que com esta renovação todas as dez heurísticas estejam presentes. Presente fica aqui a intenção de fazer uma nova avaliação heurística aquando do lançamento do “Curve Fever 3”.

Existem, evidentemente, demais jogos com características semelhantes às do “Curve Fever” que merecem igualmente uma avaliação heurística, no entanto, a presença de investigações para este jogo em específico na comunidade científica é inexistente. Deste modo, é pretendido que este artigo seja pioneiro e que sirva de contribuição nesta matéria.

Heurística	S1		S2		S3		S4		S5	
	+	-	+	-	+	-	+	-	+	-
H1										
H2										
H3										
H4										
H5										
H6										
H7										
H8										
H9										
H10										

Tabela 3. Aspetos positivos (+) e negativos (-) no geral identificados pelos sujeitos avaliadores (S)

AGRADECIMENTOS

Um agradecimento especial a Rita Albuquerque, Luna Duarte, Beatriz Pais, e Emília de Sousa e Castro pela disponibilidade em participar neste estudo.

REFERÊNCIAS

1. S. de J. Almeida, “A interação jogador e videogame na construção da experiência de jogo,” Universidade de Aveiro, 2013.
2. L. Levy and J. Novak, *Game Development Essentials: Game QA & Testing*, 1st ed. Delmar Learning, 2009.
3. K. Salen and E. Zimmerman, *The Game Design Reader: A Rules of Play Anthology*. MIT Press, 2006.
4. G. Costikyan, “I Have No Words & I Must Design,” 1994. [Online]. Available: <http://www.costik.com/nowords.html>. [Accessed: 31-May-2016].
5. B. Suits, *The Grasshopper: Games, Life and Utopia*, vol. 9. Broadview Press, 2005.
6. R. Caillois and J. G. Palha, *Os jogos e os homens: a máscara e a vertigem*. Cotovia, 1990.
7. J. Huizinga, *Homo Ludens - A study of the play-element in culture*. London, Boston and Henley: Routledge & Kegan Paul, 1949.
8. Chris Crawford, “The Art of Computer Game Design.” 1982.
9. K. Isbister and N. Schaffer, *Game Usability: Advice from the Experts for Advancing the Player Experience*. Boston: Morgan Kaufmann, 2008.
10. A. Dix, J. E. Finlay, G. D. Abowd, and R. Beale, *Human-Computer Interaction (3rd Edition)*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 2003.
11. J. Nielsen, “How to Conduct a Heuristic Evaluation,” 1995. [Online]. Available: <https://www.nngroup.com/articles/how-to-conduct-a-heuristic-evaluation/>.
12. R. Jeffries, J. R. Miller, C. Wharton, and K. Uyeda, “User Interface Evaluation in the Real World: A Comparison of Four Techniques,” in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 1991, pp. 119–124.
13. M. A. Federoff, “Heuristics and Usability Guidelines for the Creation and Evaluation of Fun in Video Games,” *Indiana University, Bloomington*.
14. D. Pinelle, N. Wong, T. Stach, and C. Gutwin, “Usability Heuristics for Networked Multiplayer Games,” in *Proceedings of GROUP 2009*, 2009, pp. 169–178.
15. T. W. Malone, “What Makes Things Fun to Learn? Heuristics for Designing Instructional Computer Games,” in *Proceedings of the 3rd ACM SIGSMALL Symposium and the First SIGPC Symposium on Small Systems*, 1980, pp. 162–169.
16. H. Korhonen and E. M. I. Koivisto, “Playability Heuristics for Mobile Games,” in *Proceedings of the 8th Conference on Human-computer Interaction with Mobile Devices and Services*, 2006, pp. 9–16.
17. H. Korhonen and E. M. I. Koivisto, “Playability Heuristics for Mobile Multi-player Games,” in *Proceedings of the 2nd International Conference on Digital Interactive Media in Entertainment and Arts*, 2007, pp. 28–35.
18. J. Paavilainen, J. Hamari, J. Stenros, and J. Kinnunen, “Social Network Games: Players’ Perspectives,” *Simul. Gaming*, vol. 44, no. 6, pp. 794–820, 2013.
19. R. Ismail, “Curve Fever Press Release.” [Online]. Available: <http://press.curvefever.com>. [Accessed: 31-May-2016].
20. G. van den Burg, “Our plans for Curve Fever 3,” 2015. [Online]. Available: <http://curvefever.com/content/our-plans-curve-fever-3>. [Accessed: 31-May-2016].

CODing theGAME: projeto de um videodocumentário sobre a cena de produção de jogos digitais em Portugal

Rogério Tavares

Arts Department

Universidade Federal do Rio Grande
do Norte – Universidade de Coimbra
rogertavares@gmail.com

Licínio Roque

Department of Informatics Engineering

University of Coimbra
3030-290 Coimbra, Portugal
lir@dei.uc.pt

RESUMO

Este trabalho é uma comunicação a fim de informar aos membros da academia sobre a produção de um vídeo documentário que está a ser realizado como parte de uma pesquisa de pós-doutoramento celebrada entre o

Departamento de Engenharia Informática da Universidade de Coimbra, Portugal, e o Departamento de Artes da Universidade Federal do Rio Grande do Norte, Brasil. Tal comunicação apresenta a pesquisa de maneira geral e foca-se na produção do documentário, parte integrante das atividades a serem realizada no período. Tal detalhamento apresenta e discute os conceitos gerais do produto apresentado, assim como sua proposta visual e métodos de produção. O objetivo é recolher um *feedback* inicial que possa aumentar a aceitação do produto final, desta maneira, o produto apresentado aqui encontra-se em evolução, e deve ser modificado durante o processo produtivo, a fim de ser apresentado no VideoJogos 2017 e demais eventos.

Palavras-chave

Documentário; Videojogos; Videogames; Projeto; Conceito

ACM Classification Keywords

Surveys and overviews, Design, Empirical studies, Biographies, General literature

INTRODUÇÃO

O documentário *CODing theGAME* faz parte do projeto de estágio de pós-doutoramento do pesquisador Prof. Dr. Rogério Tavares, do Departamento de Artes da Universidade Federal do Rio Grande do Norte, junto à Universidade de Coimbra, sob supervisão do Prof. Dr. Licínio Roque.

O projeto de pós-doutoramento trata da pesquisa sobre os processos criativos do artista de jogos portugueses. Investiga-se o percurso que as ideias criativas das artes visuais fazem durante o processo de produção de um jogo digital, e, futuramente, pode ser comparado com artistas de outras culturas. Desta maneira pode-se entender as semelhanças e

as diferenças estéticas entre os jogos criados no Brasil e em Portugal.

Por estética entendemos não apenas a visualidade, foco principal deste trabalho, mas também as questões de texto e mecânicas de jogo [6], utilizadas entre os desenvolvedores dos dois países.

A hipótese inicial do pós-doutorado, é de que, embora os jogos digitais sejam uma mídia com características universais, as produções locais mantêm características próprias de acordo com a sua cultura e seu modo de trabalho; e que mesmo países com línguas e costumes semelhantes possuem essas diferenças.

Ao entenderem-se os processos criativos, abrem-se portas para que metodologias e ferramentas sejam criadas de maneira mais apropriada às pessoas que fazem uso destas, entendendo-as individualmente, e como parte de uma comunidade de artistas que produzem artes visuais para jogos digitais.

Além dos produtos tradicionais com os resultados, como artigos e um livro, foram vislumbrados também uma exposição de arte, e publicações em meios não-científicos.

A ideia do documentário, que não integrava o projeto original, surgiu no início do trabalho, durante a abordagem de como seriam entrevistados os artistas que fariam parte da amostra. Em um breve resumo: para o plano original de trabalho são necessárias várias entrevistas com criadores de videojogos portugueses, acerca de seus processos criativos. Para isso são necessários deslocamentos, pré-entrevistas e filmagens, que não seriam divulgadas, apenas a parte transcrita de seu texto, necessária ao projeto original. Entretanto, com algumas alterações, mais algum tempo, e equipamentos e softwares melhores, esse material pode ser devidamente capturado, organizado e editado para a obtenção de um produto videográfico, alocando-se um novo prazo para a edição, e adicionando-se mais algumas entrevistas, com outros atores que compõe o ecossistema de produção de jogos digitais, além dos artistas. Desta maneira, o documentário aparece como um novo produto do projeto

original, um *spin off*, junto aos artigos, exposição, livro e divulgação, inicialmente previstos.

Entende-se que o videodocumentário e a pesquisa de pós-doutoramento são, dessa maneira, produtos diferentes, com necessidades particulares.

CODing theGAME visa cobrir exclusivamente o cenário de produção de jogos em Portugal. Brasil, e outros países de língua portuguesa, encontram-se em cenários diferentes de produção, e o documentário serve à pesquisa do pós-doutorado, ilustrando e ampliando o projeto original, que posteriormente pode ser comparado com o cenário brasileiro. Além do que, um produto semelhante em termos da extensão territorial brasileira, por si só inviabilizariam uma produção feita em um ano, por uma pessoa apenas. Acerca das fases de produção usamos os três estágios citados por Zagalo [4] em seu livro sobre a indústria de videojogos em Portugal.

PRODUÇÃO

O levantamento dos possíveis entrevistados foi aproveitado do projeto de pós-doutoramento que originou este documentário. Inicialmente 37 empresas, que após dois meses em Portugal acresceram-se mais vinte à lista, através de entrevistas preliminares e pesquisas online [7, 8]. Tais fontes passam por um processo de organização e seleção centrado em sua produção criativa. Serão, entretanto, acrescidas mais algumas entrevistas, como academia e publishers, por exemplo, de modo a vislumbrar melhor os diversos cenários da produção de jogos em Portugal. Nestas, além dos criadores previamente selecionados, incluiremos academia e indústria, como professores e coordenadores de cursos, investidores, empresários, e outras pessoas que tenham vivenciado o ecossistema em questão, ou parte dele.

As entrevistas serão gravadas nos próprios ambientes profissionais dos entrevistados, com o pesquisador deslocando-se para os devidos locais, após uma pré-entrevista realizada por e-mail ou telefone. Serão também, na oportunidade, registradas amostras de trabalhos, para serem usados como *inserts* no vídeo, ou como material para exposições, artigos, e demais produtos gerados.

Os custos de produção são arcados pelo próprio pesquisador, visto que, até o momento, não há bolsas de fomento para a produção. Busca de recursos através de sistemas de crowdfunding podem ser explorados, em se havendo tempo hábil para isso.

ROTEIRO

O gênero a ser adotado durante todo o filme é o documental, baseado em entrevistas com profissionais da área, especialmente desenvolvedores, mas com uma abertura para a história e o ensino.

O tempo de duração será curta metragem, devido ao grande número de eventos para divulgação. Sobre a duração denominada curta metragem, esta varia de acordo com as

normas de cada país. Embora no Brasil tal formato seja definido pela Medida Provisória 2.228-1 de 6 de Setembro de 2001, como “aquela cuja duração é igual ou inferior a quinze minutos” [1], a Academia de Artes e Ciências Cinematográficas dos EUA, a situa em 40 minutos [2].

Como o objetivo do documentário é mapear a produção contemporânea de jogos em Portugal, o quesito histórico será baseado no livro *Videojogos em Portugal: história, tecnologia e arte* [4] através de uma entrevista com o autor, ou na impossibilidade desta, com um resumo com locução em *off* sobre imagens dos jogos antigos.

Como serão apresentados alguns projetos desenvolvidos no ensino superior é importante apresentar entrevistas com professores ou coordenadores de cursos e disciplinas de jogos. Para estes o roteiro da entrevista é diferente de acordo com cada perfil, mas sempre com o objetivo de mapear a atual produção, e, no caso da academia, vislumbrar um futuro provável.

As entrevistas serão gravadas sempre que possível no local de trabalho do entrevistado, a não ser que se opte por outro mais representativo. Como o roteiro prevê locações em diversas cidades, serão criadas rápidas sequências de imagens, que visam contextualizar uma geografia humana da produção, assim como trazer um aspecto menos maquínico para o filme como um todo.

As respostas serão intercaladas de acordo com o tipo de pergunta, ou tópico que esteja sendo tratado. As perguntas e os tópicos podem ser vistos na seção *Questionários* a mais adiante.

A narrativa será realizada com locução em *off*, durante a maior parte do tempo, e, eventualmente pelos próprios entrevistados, em *inserts* de vídeo.

A trilha sonora será música de estoque, ou áudios dos jogos que aparecerem no documentário. Caso haja a possibilidade de investimento financeiro, músicos de games serão convidados para criar as músicas de abertura e encerramento para o filme. Existe a possibilidade da música ser criada “em casa” também, caso haja tempo viável, com instrumentos e bibliotecas Native Instruments e outras, mixados sobre áudio gravado, em softwares dedicados do tipo DAW. A introdução da sonoridade da guitarra portuguesa na trilha sonora, mesmo que misturada com sons eletrônicos, amplia o espaço humano do documentário, ao mesmo tempo que o individualiza em relação aos outros.

Embora a referência mais comum em documentários sobre produção de videogames seja o filme *Indie Game: The Movie* [5], *CODing theGAME* não terá o apelo emotivo de entrar na vida pessoal das personagens, tampouco mostrar pessoas bravas, desesperadas ou felizes, e sim mapear o país através da produção. Nesse sentido os elementos humanos serão a geografia, a música e a gastronomia, entre outras possibilidades. Pode-se dizer que as diversas culturas e turismo deixam a sua marca nesta produção.

METODOLOGIA

O workflow de produção acontecerá em 10 etapas:

1. Seleção de nomes a serem entrevistados
2. Pré-entrevista por e-mail ou telefone
3. Deslocamento do pesquisador à locação
4. Entrevista e captação de imagens
5. Pré-edição para ajustes básicos no áudio e no vídeo
6. Minutagem dos assuntos abordados
7. Publicação do material bruto
8. Armazenamento para futura edição do documentário final
9. Edição do documentário final após a finalização de todas as entrevistas
10. Pós-produção e divulgação

O cronograma foi aproveitado do projeto original, mas devido à nova série de atividades, em especial a edição, tratamento de cores, e renderização, este prazo deve ser acrescido em pelo menos 3 meses.

QUESTIONÁRIOS

Os questionários são guias para as entrevistas. De um lado eles normatizam as questões para todos, ou quase todos os entrevistados, mas devem também ser flexíveis de modo a permitir que novas questões surjam durante as entrevistas.

Pré-questionário

Existe um pré-questionário a ser respondido por e-mail ou telefone. Este tem a função de organizar e listar um número significativo de participantes. Todos os respondentes ao pré-questionário terão seus nomes listados no site ou nos créditos do filme final.

Pré-questionário

Nome, idade, natural de qual cidade e país.

Cidade na qual reside.

Em quantos jogos já participaste na produção (ensino, publicação, etc)? Considere todos os jogos.

Qual a tua função, ou funções, na produção desses jogos?

Quais desses jogos já foram publicados ou autopublicados? Em quais lojas e plataformas?

Teria interesse e disponibilidade para ceder uma entrevista a fim de participar de um documentário sobre a produção de jogos em Portugal?

Poderia me recomendar 3 pessoas que acreditas que não pudessem faltar nesse documentário?

Fim

Uma vez realizado este levantamento, segue-se para a organização das entrevistas. Estas devem ser otimizadas com base no deslocamento e datas de gravação, pois uma vez que o país inteiro será coberto, de norte a sul, a falta de uma organização dessas pode inviabilizar o prazo de produção, que é estimado de um ano a um ano e seis meses. Até o momento temos 16 cidades listadas.

Questionário

Por sua vez, o questionário completo será gravado prioritariamente no local aonde o entrevistado realiza o seu trabalho. Dessa maneira, as perguntas do pré-questionário são repetidas, com exceção da última, a fim de registrar as respostas em vídeo. É importante repetir essas perguntas em vídeo para que as respostas possam ser aproveitadas no filme, caso necessárias.

As questões são pré-estruturadas, oferecendo assim, possibilidades que sujam outras durante o decorrer da entrevista. divididas por categorias, tais quais:

Identificação:

Nesta categoria acontece uma identificação rápida do entrevistado. Alguns desses dados aparecerão apenas nos créditos e nas legendas.

Percepção:

Aqui procura-se identificar a relação do entrevistado com os jogos, assim como opções de motivação e percepção geral. Esta faz parte da pesquisa sobre processo de criação, e provavelmente não aparecerá na edição final, a não ser que constate-se algo muito relevante para o documentário como um todo.

Arte:

Nesta categoria procuramos identificar a relação do entrevistado com a arte. Como a anterior, está mais relacionada à pesquisa do que ao documentário como produto. Não será realizada para os entrevistados que não forem artistas, público-alvo da pesquisa.

Jogos Indies:

Esta categoria visa verificar se o entrevistado percebe as diferenças políticas e estéticas dos principais modos de produção atuais.

Fluxo de informação

Nesta categoria verifica-se o entendimento que o entrevistado tem do trabalho dele como indivíduo e como equipe, e se como as informações transitam e são negociadas. No quesito produtivo é importante entendermos como as equipes de trabalho funcionam internamente, e na questão estética como os conteúdos são modificados nesse processo.

Projeto

Nesta categoria busca-se entender as tomadas de decisão em relação ao projeto. Como os objetivos devem ser alcançados, como as mudanças são vistas, como a avaliação é feita, como as críticas são entendidas.

Criatividade

Aqui busca-se entender a criatividade sob os pontos de vista artístico e produtivo, e as abordagens inovadoras ou conservadoras.

Formação

Em sendo uma área relativamente nova, é comum a dificuldade em mão de obra. Uma vez que as escolas estão formando pessoas nessa área é importante saber se o mercado e a academia estão dialogando ou se enxergam as necessidades de uma maneira discordante.

Identidade

Nesta categoria buscamos entender o que é um jogo português, no que ele se diferencia dos outros e se os envolvidos tem esse tipo de preocupação ou acham mais importante manter-se em padrões pré-estabelecidos. Busca-se também averiguar se há um reconhecimento dentro da indústria por esse tipo de pensamento.

Políticas

Finalmente, procuramos sondar o nível de engajamento político dos profissionais envolvidos nesse ecossistema. O que eles consideram importante neste momento, e como eles conseguem enxergar essa indústria nos próximos anos.

Gravações

No total são 30 perguntas, e o tempo estimado da entrevista pode variar de uma a duas horas, dependendo do entrevistado. Algumas, baseadas no gosto pessoal ou na experiência são mais rápidas. Outras demandam algum pensamento. Por isso o questionário pode ser enviado antes da gravação, caso o entrevistado assim o prefira.

CONCEITO E VISUALIDADE

Buscamos representar o conceito do documentário em seu logotipo. Para tanto o logotipo apresenta duas tipografias distintas e um elemento pictórico.

O logotipo do documentário é apresentado em 4 variações, para uso com fundos claros e escuros, e para uso na



Figura 1: Logotipo em quatro aplicações, para fundo claro e escuro, com legenda ou sem.

horizontal e na vertical. As opções horizontais contém o subtítulo do filme.

As variações verticais são aplicação em materiais com espaço limitado, e junto a outros logotipos. As horizontais são para aplicações em documentos e materiais com mais espaço informativo.

O elemento pictórico baseia-se na estilização de uma peça de bacalhau como as que são encontradas à venda em mercados. O rabo foi ligeiramente ampliado para ser reconhecido como um peixe grande, visto que depois de seco o rabo perde um pouco de seu volume. O teste de visualização quando feito com 3 brasileiros, todos reconheceram um peixe grande sem cabeça, embora apenas um fizesse a conexão com um bacalhau. Quando feito com dois portugueses, por seus conteúdos culturais, reconheceram o bacalhau imediatamente.

O elemento textual é um jogo de linguagem com a palavra Cod, bacalhau em língua inglesa, e sua aparição na palavra Code, código na mesma língua. Nesse texto, o ato de codificar, Coding, internaliza o peixe, em referência a um dos pratos símbolos da cultura portuguesa, internacionalmente reconhecido. Ou seja: como entender o jogo, elemento universal, a partir de uma ótica local.

Simbolicamente, a pesca do bacalhau remonta ao Século XIII, quando os portugueses já dominavam a pesca desse alimento. Essa pesca era feita com vara, ou seja, um a um, e pescava-se em equipes, nunca sozinho, e as iscas preferidas eram as lulas. [3]

Desta maneira faz-se uma aproximação simbólica nas quais os jogos são produzidos um a um, por times. Embora também possam ser produzidos em série, ou individualmente, mas essas produções, de maneira geral, assemelham-se mais a pequenos aplicativos do que a jogos, que em geral são produzidos por equipes interdisciplinares. O investimento em uma isca cara, também é notável, desde que é necessária uma pesca anterior, a da lula, para em seguida alcançar-se um peixe maior. Isso demonstra projeto, empreendedorismo, e visão de futuro, características seculares, presentes até hoje. Do contrário, o povo português seria comedor de lulas, e não de bacalhaus.

A tipografia adotada constitui-se nas fontes Nonesuch, de Tyler Finch,¹ e Flawless Flygirl, de Phil Bracco.² Embora ambas sejam fontes proprietárias, são de baixo custo, e trazem o frescor de fundições independentes.

A fonte Nonesuch utilizada no logotipo, representa a institucionalização dos jogos enquanto produtos de grandes empresas, que se sustentam apenas com vendas muito representativas. Reconhece-se que não seja a fonte ideal,

1 <https://creativemarket.com/finck/609076-Nonesuch>

2 <https://creativemarket.com/PBroccoli/839384-Flawless-Flygirl>

devido a altura do pingo do “i”, que não proporciona a quebra esperada entre a palavra Cod e o gerúndio de codificar. As fontes gratuitas em sua maioria fracassaram no espaçamento entre as letras maiúsculas de COD, outras não apresentavam minúsculas ou caracteres acentuados, que podem ser necessários futuramente, e outras encontravam um custo muito alto para a fase inicial de produção, quando foi necessário o investimento em outros softwares e equipamentos. Pode acontecer durante o processo a substituição da fonte por outra mais apropriada como, a Kapra,³ que, embora bem mais cara, possui mais opções aumentando as possibilidades. Como este projeto está sendo realizado sem apoio financeiro, as economias, mesmo em poucos dólares, são necessárias.

A fonte Flawless Flygirl, utilizada na legenda “videogame makers in Portugal”, por sua vez representa o processo artesanal de se fazer um jogo. Por detrás dos milhares de cópias vendidas existe um processo inicial, bastante manual, que inicia-se com processos criativos que tomam forma em textos e desenhos, muitas vezes feitos pixel a pixel, centenas de testes de funcionamento e também de aceitação, e chega a utilizar-se inclusive de nomenclaturas como produção, testes e polimento, remetendo a ambientes como oficinas.

A utilização de ambas as fontes nos logotipos pode ser vista na Figura 1, enquanto que a fonte Flawless Girl, será usada também nas *video captions*, como no exemplo da Figura 2.

Esperamos que esta comunicação visual agrade e represente o espírito de um povo aventureiro, que possui uma população jovem a tentar se colocar no mundo como criadores de jogos relevantes, tal como seus antepassados lançavam-se ao mar.

PERSPECTIVAS FUTURAS

Como este é um projeto ainda em desenvolvimento, algumas das características apresentadas, em termos de produção, podem ser alteradas. Entretanto, o objetivo é sintético e perene: mapear a produção contemporânea de videogames em Portugal, durante os anos 2016-17, e divulgar esse cenário através de um curta metragem que possa ser exibido em festivais e disponibilizado na internet.

AGRADECIMENTOS

O pesquisador agradece ao supervisor Prof. Dr. Licínio Roque e seus pesquisadores por receberem-no em vossa instituição, e à Universidade Federal do Rio Grande do Norte pelo afastamento concedido a fim de realizar esta pesquisa, e os produtos resultantes dela, como o documentário acima descrito. Agradece-se também a todos



Figura 2: Aplicações das fontes como selo e como legenda (*video caption*).

os participantes envolvidos no processo até agora, entrevistados, professores, empresários, pareceristas e a todos que colaboraram com críticas e sugestões.

REFERENCES

1. Presidência da República, Casa Civil, Subchefia para Assuntos Jurídicos, Medida Provisória N° 2.228-1, de 6 de Setembro de 2001.
http://www.planalto.gov.br/ccivil_03/mpv/2228-1.htm (Acessado em 04/08/2016)
2. http://www.oscars.org/sites/oscars/files/89aa_short_subject.pdf
3. Pesca do bacalhau:
https://pt.wikipedia.org/wiki/Pesca_do_bacalhau_pelos_p_ortugueses
4. ZAGALO, N. (2013) Videogames em Portugal: história, tecnologia e arte. FCA; Lisboa, Portugal.
5. Indie Game: the movie. (2013). Direção de Lisanne Pajot e James Swirsky; com entrevistas de Jonathan Blow, Phil Fish, Edmund McMillen. Documentário, colorido, 1h34min.
6. GALLOWAY, A. R. (2006). *Gaming: Essays on Algorithmic Culture*. (S. Weber, M. Poster, & N. K. Hayles, Eds.) Choice Current Reviews for Academic Libraries (Vol. 18). University of Minnesota Press.
7. Game Dev Map <https://www.gamedevmap.com>
8. Portugal Game Studios Spotlight 2015.
<http://bit.ly/2eG5luP>

3 <https://creativemarket.com/blezja/102239-Kapra-fonts-family>

Exploração do High Fidelity para desenvolvimento de jogos em mundos virtuais imersivos multiutilizador

José Martins

INESC TEC e Universidade
Aberta
Lisboa, Portugal
martins.josemanuel@gmail.com

Leonel Morgado

INESC TEC e Universidade
Aberta
Coimbra, Portugal
leonel.morgado@uab.pt

Vítor Cardoso

CIAC e Universidade Aberta
Lisboa, Portugal
vitor.cardoso@uab.pt

RESUMO

Apresentamos o resultado de uma exploração prática da tecnologia de mundos virtuais imersivos multiutilizador High Fidelity, baseada em tecnologia Web. Esta tecnologia permite criar mundos virtuais cujas formas de interligação, controlo e interação tiram partido de uma Interface de Programação de Aplicações em JavaScript. Através do caso prático de desenvolvimento de um protótipo de jogo educativo simples, descrevemos a tecnologia High Fidelity, incluindo o tipo de *scripts*, a arquitetura inerente e as suas características de desenvolvimento e utilização. Destacamos as dificuldades inerentes ao estado atual da plataforma, em constante reformulação e algumas peculiaridades.

Palavras-chaves do autor

Mundos virtuais; simulações; jogos; High Fidelity

INTRODUÇÃO

A utilização de simulações e/ou jogos nas empresas e outras organizações nos sistemas de gestão de formação, baseada em tecnologia Web, vem sendo desenvolvida há vários anos, em áreas tão diversas como a manutenção mecânica aeronáutica, condução automóvel, controlo de submarinos, fisioterapia remota ou ciclismo [1]. A tecnologia High Fidelity é uma proposta recente nesta área, que recebeu alguma atenção mediática por provir de uma empresa fundada pelo fundador do Second Life, Philip Rosedale. O High Fidelity posiciona-se como uma tecnologia aberta, que instalável em computadores pessoais e que disponibiliza uma interface para programação de aplicações (API, *Application Programming Interface*) em JavaScript, visando permitir a qualquer pessoa com conhecimentos técnicos criar e partilhar um ambiente de realidade virtual.

É uma tecnologia recente, encontrando-se disponível em <https://highfidelity.io/> [acedido a 10 de outubro de 2016] à data do presente artigo ainda em versão de testes. Aliás, durante a exploração aqui descrita, o ritmo de lançamento de versões era praticamente diário. Assim, salienta-se que aquando da leitura do presente artigo a tecnologia poderá

ter sofrido alterações.

Contudo, o desenvolvimento para High Fidelity defronta-se com faltas de documentação várias. Aspetos aparentemente cruciais, como a localização de ficheiros de *backups* ou de configuração, não são descritos na documentação disponível, o que dificulta sobremaneira a utilização desta tecnologia. Desta forma, o resumo aqui apresentado, resultado da nossa experimentação ao longo de alguns meses, poderá contribuir para que mais pessoas possam tirar partido dela e explorá-la.

HIGH FIDELITY- BREVE DESCRIÇÃO

Conforme já se mencionou, o High Fidelity é uma tecnologia de mundos virtuais imersivos multiutilizador. A empresa que a produz designa por “metaverso” o conjunto desses mundos virtuais, opção comum na terminologia da área [11]. Cada mundo criado com o High Fidelity, porção finita do metaverso, é designado por “domínio”. A cada domínio pode-se associar um *place name* que o identifica no metaverso do High Fidelity [6]. Pode ainda ter-se domínios isolados do metaverso, designados por domínios locais. Conforme a configuração o acesso a um domínio pode estar aberto a qualquer utilizador, inclusivamente para editar conteúdos e adicionar entidades (configuração predefinida), ou ser restrito em maior ou menor grau [8].

Cada utilizador interage com um domínio (ou todo o metaverso) do High Fidelity através de um programa cliente, que efetua a ligação aos servidores e produz a representação visual do espaço. Os utilizadores, como é habitual neste tipo de plataformas, são representados como parte integrante do mundo virtual, sob a forma de um avatar. No HighFidelity, este programa cliente designa-se “Interface”, disponibilizando formas de interação habituais, como rato, teclado e ecrã ou mais imersivas como óculos de realidade virtual (*head-mounted displays*) ou controladores espaciais para as mãos.

O programa cliente Interface disponibiliza também ferramentas para manipular o domínio, o seu conteúdo e para executar *scripts* em JavaScript. O conteúdo consiste em modelos tridimensionais, que podem ser importados nos formatos .fbx ou .obj; e em *scripts*, com particularidades de alojamento e execução que se descrevem na secção seguinte.

Supported by research effort FourEyes of project TEC4Growth. “TEC4Growth - Pervasive Intelligence, Enhancers and Proofs of Concept with Industrial Impact/NORTE-01-0145-FEDER-000020” is financed by the North Portugal Regional Operational Programme (NORTE 2020), under the PORTUGAL 2020 Partnership Agreement, and through the European Regional Development Fund (ERDF).

SCRIPTS NO HIGH FIDELITY

Os *scripts* suportados pelo High Fidelity são escritos em JavaScript. Há uma particularidade: além de poderem estar alojados no servidor do domínio HighFidelity, podem estar alojados em qualquer servidor com tecnologia Web, inclusivamente em serviços de *cloud* públicos. E podem ser executados de forma distribuída: no programa cliente, no servidor de domínio ou noutros servidores, especificamente para execução de *scripts* [10]. Dividem-se em três categorias: *scripts* de interface, *scripts* de entidades e *scripts* de clientes de atribuição [9].

Os *scripts* de interface permitem personalizar a experiência de interação do utilizador num domínio, alterando o funcionamento do programa cliente Interface. Por exemplo, através da criação de novos menus e de acessórios. Ou seja, permitem personalizar a interface de utilizador com aspetos pessoais, inacessíveis e não visíveis aos outros visitantes do domínio. Podem, contudo, afetar o domínio. Se um *script* de interface afetar o domínio, o resultado será visto por todos os programas clientes, que são informados da alteração pelo servidor do domínio HighFidelity. Os *scripts*

de interface são executados pelo programa cliente Interface, através de um *script* próprio, designado Runner Script. A sua execução termina quando se fecha o programa cliente Interface.

Os *scripts* de entidades são associados às entidades do domínio, entendidas como objetos e restante conteúdo visual do espaço tridimensional. São usados para permitir interações com as entidades ou executar alguma ação automática. Este tipo de *scripts* é pré-carregado pelo programa cliente Interface quando o utilizador se aproxima da entidade e descarregado quando se afasta dela, pois são executados concorrentemente pelos programas clientes dos utilizadores. Qualquer alteração que ocorra na entidade em consequência da execução de um *script* de entidade é reportada pelo programa cliente ao domínio, o que origina a sua visualização nos programas clientes dos demais utilizadores presentes nesse domínio. Estes *scripts* de entidades podem ter o código alojado no servidor do domínio (por inserção direta do código nas propriedades da entidade) ou ter o código alojado em servidores na Web (indicando URL do código do *script* nessas propriedades).

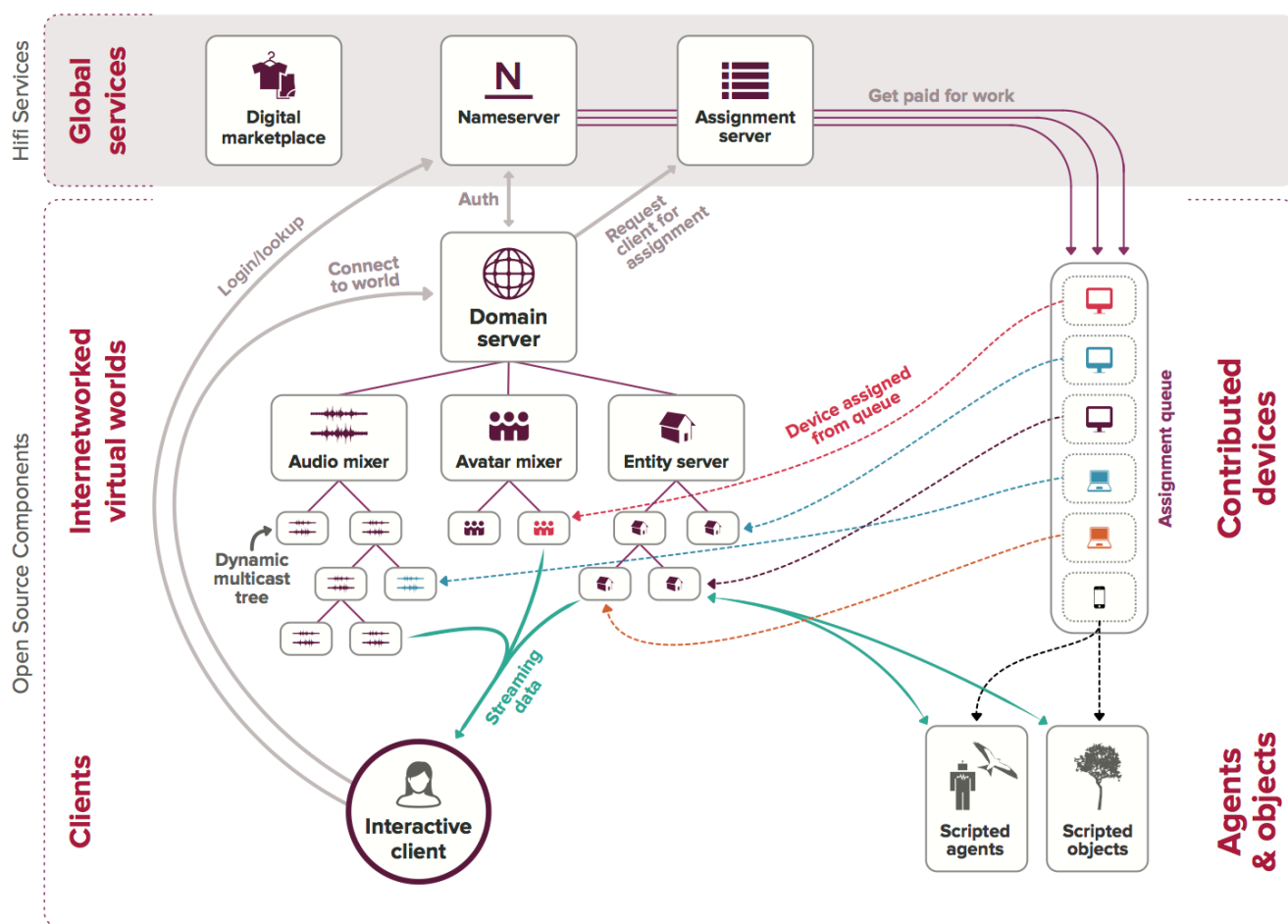


Figura 2. Visão na terceira pessoa com âmini-espelho.

Fonte: https://files.readme.io/fE5d8oNxSaQPIUSa7Qs3_bfkWsEouTfGTsG3PEHV.png [10 de outubro de 2016].

Esta última opção não estava disponível no momento em que escrevemos este artigo [2], apesar de ter estado disponível em versões alfa durante o nosso período de exploração.

Os *scripts de clientes de atribuição* são *scripts* cuja execução está a cabo de servidores externos ao domínio. Permitem coordenar o estado do mundo virtual, jogo ou aplicação entre várias entidades, ou entre estas e avatares, além de permitir distribuir a carga de processamento. Esses servidores de execução de *scripts* são designados “clientes de atribuição” ou “agentes”. Estão ligados ao servidor de domínio e os resultados da execução são comunicados por este a todos os programas cliente (Interface). Assim, os *scripts* de clientes de atribuição são executados desde que o servidor de domínio esteja em funcionamento. Os URL dos servidores de atribuição são registados nas configurações do domínio.

ARQUITETURA DO HIGH FIDELITY

Um domínio é gerido por uma pilha de quatro servidores virtuais que tem no topo o Servidor de Domínio, como ilustra a Figura 2. O Servidor de Domínio é o principal, sendo iniciado para criar o mundo virtual. É responsável por fornecer o seu número IP aos clientes Interface, para comunicação por parte destes; e por distribuir tarefas de simulação aos outros servidores: serviços globais (nomes e atribuições); Misturador de Áudio; Misturador de Avatares; e Servidor de Entidades. Estes servidores estão normalmente na máquina que aloja o domínio, mas podem estar noutra [6].

O Misturador de Avatares é responsável por tudo o que esteja relacionado com presença dos avatares no domínio: posição, modelo, movimentação, expressões faciais, etc. O Misturador de Áudio mistura os sons do ambiente com os dos avatares, apresentando uma mistura a cada avatar de acordo com a sua posição em relação às fontes sonoras. O Servidor de Entidades controla todas as entidades do domínio, que têm *scripts* associados. Qualquer alteração numa entidade do domínio é-lhe comunicada, que a retransmitirá para todos os clientes Interface.

Dos serviços globais, o Servidor de Nomes permite encontrar domínios e locais específicos dentro de um domínio, através de nomes únicos. Permite ainda autenticar utilizadores: quando um domínio é criado, o acesso ao mesmo pode ser restringido a um conjunto de utilizadores, cabendo ao servidor de nomes verificar as identidades [7]. O Servidor de Atribuições permite registar no High Fidelity um dispositivo computacional na Internet para executar tarefas: servidor de domínio ou serviços de execução de *scripts*. O Servidor de Atribuição delega tarefas aos dispositivos disponíveis aos utilizadores (*ibid.*).

INSTALAÇÃO E CONFIGURAÇÃO

A tecnologia High Fidelity encontra-se disponível para Windows, Mac Os e Linux. O modo mais simples de

configurar um domínio é através da aplicação *Sandbox*, disponível para Windows e Mac OS. Também é possível compilar o servidor domínio e os clientes de atribuição a partir do código fonte disponível.

A *Sandbox* cria um domínio *Home* na máquina local (<http://localhost:40100>). Quando se encontra em execução inicia o servidor de domínio e disponibiliza um ícone na barra de tarefas. Neste, acede-se a um menu onde a opção *Go Home* executa o cliente Interface neste domínio *Home*, com o avatar na posição configurada para o domínio, por predefinição as coordenadas (0,0,0). Apesar do nome, este domínio pode ser acedido por outros utilizadores na rede, se as configurações de rede o permitirem.

Quando a *Sandbox* é executada pela primeira vez, é atribuído ao domínio *Home* um *place name* temporário no servidor de nomes. O novo domínio, para ser reconhecido no metaverso no High Fidelity, deve estar associado a uma conta no site do High Fidelity, configurada na opção *Metaverse/Networking* do menu.

Pode-se restringir o acesso ao domínio a um utilizador específico, bem como limitar a capacidade máxima de utilizadores, as permissões destes e a largura de banda de envio de dados para cada utilizador.

A natureza distribuída do High Fidelity permite ter os recursos num servidor público na Internet e/ou no servidor de recursos do domínio, que se ativa com a opção *Asset-server* da configuração. Nesta situação, detetámos que os recursos locais em servidores Windows são alojados por em `<PastaDoUtilizador>\AppData\Roaming\High Fidelity\ assignment-client\assets\files` (não testámos a localização noutros sistemas operativos).

Em relação aos *backups*, os mesmos podem ser configurados seleccionando a opção *Entity Server Settings*. Neste item, entre outras configurações, é possível definir vários tipos de *backups* em termos de intervalos de tempo e número máximo de versões; o tempo máximo de vida para a entidades temporárias; e o caminho para o ficheiro com os *backups*. Os *backups* são guardados em `<PastaDoUtilizador>\AppData\Roaming\High Fidelity\ assignment-client\entities`.

A importação dos *backups* é efetuada através do cliente Interface, na opção *Import Entities* do menu *Edit*. Esta só se encontra disponível se a opção *Advanced Menus* do menu *Settings* tiver sido ativada. Para restaurar um domínio através do ficheiro do *backup*, é necessário que este esteja descompactado. Além disso, só são aceites para importação ficheiros com dados em formato JSON.

DOMÍNIO EXEMPLIFICATIVO: PROTÓTIPO DE JOGO

Descrição do Domínio

A exploração que permitiu efetuar a descrição aqui efetuada foi efetuada no decurso da criação de um domínio exemplificativo no High Fidelity, que simula um jogo sério

para aprendizagem de matemática, nomeadamente sólidos geométricos. O espaço de aprendizagem restringe-se ao interior de uma casa, numa região de montanha. O exterior da casa é caracterizado por um ambiente de montanha com algumas árvores, iluminado ao longo do dia pelo percurso do sol (Figura 5). No interior desta pode-se encontrar uma secretária com livros, uma folha com a planificação do cubo, um candeeiro a óleo, uma cadeira, um fogão com uma caixa de fósforos, um fósforo, sólidos geométricos espalhados pela casa, duas caixas, uma mochila, uma página *web* com conteúdos sobre sólidos geométricos e outros elementos decorativos.

O objetivo é que o utilizador, após estudar sólidos geométricos numa página *web*, arrume os sólidos geométricos dispersos pela casa nas caixas correspondentes, identificadas como “Poliedros” e “Não poliedros”.



Figure 5. Exterior do domínio ilustração.

Interações

Os elementos do domínio que permitem interação (agarrar e mover) por possuírem características físicas, são: mochila, sólidos geométricos, livros, cadeira, folha com a planificação, candeeiros, caixa de fósforos e o fósforo.

A página *web* contém os conteúdos necessários a adquirir para organizar os sólidos na caixa correta, Figura 6. Quando um sólido é colocado na caixa correspondente o nome do sólido é apresentado no quadro da caixa. Quando no interior da caixa estiverem todos os sólidos correspondentes que estavam espalhados pela casa (Figura 7) e apenas os correspondentes, é emitido um som informativo de resposta correta.

Ao clicar com a roda do rato na folha com a planificação do cubo que se encontra sobre a secretária é exibida uma nota informativa sobre o que fazer com essa folha, que poderia ser usada como pista para novas tarefas.

Dependendo do momento do dia em que a casa é visitada, poderá ser necessário usar o candeeiro a petróleo. Este possui um regulador de luz com seis níveis de intensidade. Para aumentar tem que se clicar no regulador com o lado esquerdo do rato e para diminuir tem que se clicar com a roda do rato (Figura 8).



Figure 6. Página web embebida no domínio.



Figura 7. Sólidos nas caixas correspondentes.

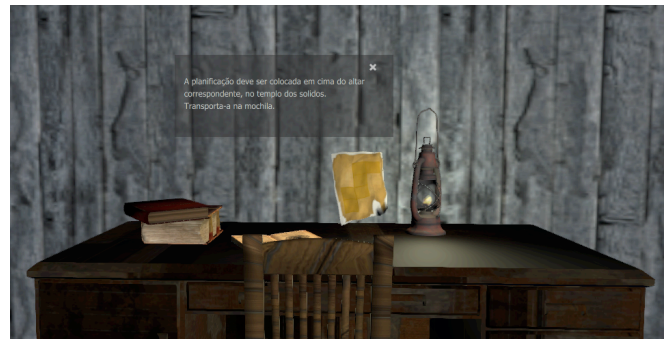


Figura 8. Interações com o candeeiro e com a folha com a planificação.

EXPERIÊNCIA DE UTILIZAÇÃO DO HIGH FIDELITY

Basta dispor de dispositivos tradicionais de interação pessoa-computador (teclado, rato e monitor), mas o High Fidelity suporta controladores somáticos (gestos das mãos) e óculos de realidade virtual.

O cliente Interface vem por predefinição com um conjunto de *scripts* configurado para execução automática, que determinam aspetos da experiência do utilizador e disponibilizam ferramentas para criação de um domínio. Pode dar-se o caso de não arrancarem e é necessário verificar a sua execução: *progress.js* (evolução do carregamento das entidades inseridas no domínio por URL); *grab.js* (permite agarrar com o rato entidades móveis); *users.js* (lista de utilizadores online e controlo da

visibilidade do utilizador no metaverso do High Fidelity); `handControllerGrab.js` (permite agarrar as entidades móveis com controladores somáticos); `notifications.js` (gera notificações no cliente Interface); `dialTone.js` (emite um som quando se liga ou se desliga do servidor); `squeezeHand.js` (orienta a animação das mãos com base no controlador somático); `handControllerPointer.js` (permite controlar o rato com o controlador somático); `examples.js`, (acesso à biblioteca de exemplos); `selectAudioDevice.js` (menu *Audio* do cliente Interface); `defaultScripts.js` (iniciação de todos estes *scripts* enumerados) [4].

Para navegar no metaverso do High Fidelity, utilizando o cliente Interface tem de se começar por definir o mundo virtual (domínio) a explorar, através do menu *Navigate*. Este permite também aceder a um local específico dentro de um domínio. O destino é dado através do respetivo *place name* (registado no servidor de nomes), endereço IP ou caminho para um determinado local.

O avatar é o representante do utilizador no mundo virtual. No cliente Interface, o menu *Avatar* permite configurar o tamanho do avatar e anexar elementos/modelos através do seu endereço URL a diferentes partes do corpo (Figura 4). Outros aspetos, como atribuir um nome ao avatar, definir o som de colisão do avatar com outras entidades e escolher a aparência, estão no submenu *Avatar*, do menu *Settings*.



Figura 4. Avatar com elementos anexados.

De referir que algumas das funcionalidades dos menus da interface, indicadas na documentação, apenas estão disponíveis aquando a ativação da opção *Advanced Menu* do menu *Settings*.

A exploração inicial do High Fidelity foi muito instável. Evidenciou dificuldades nos carregamentos dos domínios, visualização e interação com as entidades, assim como visualização do avatar. Contudo, as mesmas foram atenuadas com atualizações posteriores. Também há que destacar a instabilidade no momento de guardar o trabalho realizado. O High Fidelity guarda as alterações no domínio automaticamente, podendo-se definir diferentes tipos de

backups nas configurações do domínio. Não é referido na documentação oficial o local dos *backups* nem como os restaurar. Assim, a informação dada sobre *backups* que prestámos na seção “Instalação e Configuração” foi obtida através da consulta de fóruns e exploração da máquina de instalação. Foi experienciado que em algumas máquinas os *backups* podem não ser realizados de todos, perdendo-se o domínio assim que se desligue o servidor. A abordagem a que recorremos para contornar esta situação foi criar um *script* que carregasse e configurasse todas as entidades do domínio, o que assumimos é pouco prático e de morosa manutenção. Por exemplo, é preciso posicionar e configurar o modelo no domínio e posteriormente colocar esses dados no código do *script*. Também não está documentada a forma de repor *backups*, tendo sido constatado que é possível fazê-lo através da importação de entidades, do menu *Edit*.

A deteção e verificação de erros e anomalias nos *scripts* de entidades também não é uma tarefa fácil no High Fidelity. Sempre que se pretende efetuar uma alteração num *script* de entidade, há que reiniciar o domínio para que o mesmo seja atualizado. Os erros são reportados no *log* do programa cliente Interface, misturados com os demais registos do Interface. Apesar deste programa cliente disponibilizar um editor de *scripts*, este não permite associá-los a uma entidade.

Foram detetados alguns problemas na inserção de texturas em modelos, bem como na inclusão de invólucros convexos para as colisões entre entidades. Os modelos inseridos com texturas separadas do modelo nem sempre apareciam. Uma forma encontrada para ultrapassar este problema foi incorporar as texturas no ficheiro do modelo aquando da exportação do modelo para o formato *.fbx*. Em relação aos invólucros convexos obtidos recorrendo ao software V-HACD, que por vezes surgiam no High Fidelity desproporcionais em relação ao modelo e apresentavam frechas, deixando passar o avatar ou outra entidade. A solução passou por testar configurações do V-HACD e/ou adicionado elementos geométricos ao invólucro do modelo.

Em algumas situações deparámo-nos com alterações aos nomes das propriedades das entidades em JavaScript (<http://jsref.docs.highfidelity.com/v1.0/docs/entity-properties> [acedido a 10 de outubro de 2016]). A solução encontrada foi criar a entidade no domínio na posição e configuração pretendida e posteriormente exportá-la em formato JSON.

CONCLUSÃO

A tecnologia High Fidelity é bastante promissora, mas ainda em fase de testes, com grande instabilidade de execução e falta de documentação. Tal acrescentou dificuldades e contrariedades na concretização do exemplo aqui apresentado, mas provou ser fonte rica de descoberta: constantes atualizações do software, sobretudo na versão alfa; *bugs*; problemas com os *backups*; problemas com

texturas dos modelos e com os invólucros convexos para colisões; atualização dos *scripts* de entidades no domínio; detecção de erros dos *scripts* no *log* do cliente Interface.

Há, por isso, ainda muito para explorar no High Fidelity. Como pontos interessantes a explorar no futuro, sugerem-se a análise do funcionamento dos *scripts* de atribuição e sua interação com os *scripts* de entidades; as interações específicas para o uso de controladores de mãos e de óculos de realidade virtual; a análise do que sucede quando vários avatares pretendem interagir com a mesma entidade, provocando situações de contenção de recursos; exploração da comunicação do High Fidelity com um servidor externo para, por exemplo, definir e controlar ações ou obter e atualizar informações numa base de dados.

REFERÊNCIAS

- [1] A. Gaspar, “Simulação @ INESC TEC,” 15 fevereiro 2015. [Online]. Disponível: http://www.afceaportugal.pt/2015/eventos/A_MS260_Seminar_INESCP.pdf [acedido a 10 de outubro de 2016].
- [2] “Attaching a Script to an Entity,” High Fidelity, 2015. [Online]. Disponível: <https://readme.highfidelity.com/docs/attaching-a-script-to-an-entity> [acedido a 10 de outubro de 2016].
- [3] “Building Content,” High Fidelity, 2015. [Online]. Disponível: <https://readme.highfidelity.com/docs/getting-started-1> [acedido a 10 de outubro de 2016].
- [4] “The Default Scripts,” High Fidelity, 2015. [Online]. Disponível: <https://readme.highfidelity.com/docs/the-default-scripts> [acedido a 10 de outubro de 2016].
- [5] High Fidelity Inc, “EntityItemProperties,” 2015. [Online]. Disponível: <http://jsref.docs.highfidelity.com/docs/entity-properties> [acedido a 10 de outubro de 2016].
- [6] “High Fidelity,” High Fidelity, [Online]. Disponível: <https://readme.highfidelity.com/docs/architecture-overview> [acedido a 10 de outubro de 2016].
- [7] P. Rosedale, “High Fidelity System Architecture,” 24 abril 2014. [Online]. Disponível: <http://blog.highfidelity.com/blog/2014/04/high-fidelity-system-architecture> [acedido a 10 de outubro de 2016].
- [8] “Server Settings - reference guide,” High Fidelity, [Online]. Disponível: <https://readme.highfidelity.com/v1.0/docs/server-settings-reference> [acedido a 10 de outubro de 2016].
- [9] “Scripts Overview,” High Fidelity, 2015. [Online]. Disponível: <https://readme.highfidelity.com/docs/scripts-digging-deeper> [acedido a 10 de outubro de 2016].
- [10] “Welcome – High Fidelity Documentation,” High Fidelity, 2015. [Online]. Disponível: <https://readme.highfidelity.com/docs/contributing-to-docs> [acedido a 10 de outubro de 2016].
- [11] L. Morgado, 2009. Interconnecting virtual worlds, Journal of Virtual Worlds Research 1 (3): 4-7.

Fan culture e mods: a dimensão do multiverso cultural dos games na internet

André Pequeno dos Santos
Universidade do Porto
pequenoroyale@gmail.com

José Manuel Azevedo
Universidade do Porto
jmpazevedo@gmail.com

RESUMO

Comunidades de *mods*, produções coletivas wikia, canais de vídeos de análise, *gameplay* e notícias no YouTube e Twitch. Estas são apenas algumas das facetas com as quais os jogadores se podem envolver hoje para além da simples participação nos jogos digitais. Ao mergulhar nestas realidades alternativas e exercer o princípio do fan, estes mesmos jogadores auxiliam na manutenção e desenvolvimento de um universo cultural alternativo, partilhado e que muitas vezes se apropria do universo de produção cultural tradicional. Este ensaio tem como objetivo discutir o universo alternativo da *fan culture* nos *games*. A partir das comunidades Steam Workshop e Nexus Mods, e com a apresentação de três *mods* da série de RPG The Elder Scrolls (Helgen Reborn, Agent of Righteous Might, Falskaar), pretendemos analisar os pontos de intersecção em que as realidades da produção tradicional e alternativa se misturam, criando assim o que pretendemos chamar de um multiverso cultural.

Palavras-Chave

Comunidades de *mods*; *fan culture*, multiverso cultural; games e narrativa.

INTRODUÇÃO

O impacto sociocultural da internet encontra, atualmente, alguns elementos únicos. Dizemos únicos não tanto pela sua exclusividade, mas sim pela forma peculiar pelas quais as dinâmicas e relacionamentos se transformam quando passam para o universo digital. Pois o que dizer, por exemplo, do fenómeno da pirataria, que em questão de anos ajudou a modificar boa parte da cadeia de produção e disseminação cultural e informacional? E das relações sociais, ampliadas e reinventadas com o surgimento quase que instantâneo e omnipresente das redes sociais, personalizadas ao gosto e as necessidades dos seus utilizadores? Pois bem, o tecido sociocultural foi modificado de forma radical pela popularização da internet enquanto universo de mediação, e sob esta condição, viu rejuvenescer, ampliar e até mesmo desenvolver novos universos de produção e disseminação cultural. Sob esta nova roupagem, os estudiosos da internet, entre eles Castells (2012), Cardoso (2006), Van Dijck (2012), Gere (2008), Raine e Wellman (2014) se debruçaram na tentativa de compreender os fenómenos sociais e culturais a partir das dinâmicas de interação e tensões produzidas entre

utilizadores, instituições e organizações. A eles, se somaram outros como Jenkins (2006a) e Hills (2005), que desenvolveram estudos de comunidades alternativas, deslocadas do seu meio natural para explorar as novas possibilidades desencadeadas pela liberdade do ambiente digital. É neste domínio no qual pretendemos apresentar este ensaio, identificando estas comunidades alternativas por ambientes de *fan culture*, em que os fans são parte fundamental no consumo, preservação, disseminação e agora, ultrapassada a barreira da apropriação dos meios de produção e dos canais de distribuição, agentes ativos no ciclo produtivo. Para tanto, vamos situar este universo da *fan culture* e sua relação a partir das comunidades de *mods*, um dos estratos da cultura *gamer* na qual os jogadores se tornam produtores de conteúdos alternativos que expandem a experiência de jogo. Posteriormente, nosso argumento irá se nortear pelo conceito de «multiverso cultural», apropriado de uma das manifestações mais caras ao universo da *fan culture*, a da banda desenhada.

AS COMUNIDADES DE MODS E DE RE-PRODUÇÃO DE CONTEÚDOS

As comunidades *online* de *mods* são parte de uma das experiências de produção cooperativas mais ricas e efervescentes na atualidade. Idealmente inseridas no que Toffler (1989) e Ritzer, Dean, Jurgenson (2012) configuram como *prosumer*, conceito relativo a junção dos papéis de produtor e consumidor num só sujeito, e que posteriormente foi atualizada por Bruns (2009) com o *produser*, estas comunidades sintetizam a dupla função de alguns dos seus membros enquanto agentes de produção, ou em muitos casos, de remixagem, da re-produção de conteúdos e de sujeitos ativos nas dinâmicas de interação no seio destes ambientes, em que esta dupla função também se torna fundamental para o fechamento dos ciclos de produção que se desenvolvem neste estrato alternativo.

Estando vinculadas ao universo da *fan culture*, estas comunidades são ambientes nos quais a apropriação dos conteúdos toma outra proporção a partir do grau de complexidade de algumas das atividades desenvolvidas pelos seus membros, nomeadamente nas formas de manipulação dos conteúdos e nas formas de interação entre os sujeitos. Hong e Chen (2014), por exemplo, tratam das relações do trabalho cooperativo a partir dos *mods*, destacando a necessidade de se introduzir a questão intensiva nas produções através dos esforços realizados, do

tempo gasto e da afetividade com o qual se veem envolvidos.

Sotamaa (2010), por seu turno, busca compreender as motivações e as práticas envolvidas nas atividades de *modding*. Sua abordagem, vista a partir de um estudo conduzido no jogo Operation Flashpoint, acaba por identificar perfis distintos de *modders*, o que, consequentemente, torna inviável a ideia destes sujeitos vistos como um grupo com interesses e objetivos homogêneos.

Poor (2013) em estudo até então inédito, partindo de uma abordagem quantitativa, nos revela que os *modders*, para além de um senso de comunidade e pertença, que se revela pelo espírito cooperativo da sua relação, estão inseridos em uma faixa de idade ampla, que varia de jovens adolescentes até adultos, sendo indivíduos que não se restringem a apenas um jogo ou série, tendo no seu catálogo de *mods* os mais variados jogos.

Já a presente pesquisa centrou-se na análise de duas comunidades, provavelmente as mais populares e conhecidas na internet: A Steam Workshop e a Nexus Mods. Estes são ambientes em que tanto os produtores de conteúdos alternativos quanto os consumidores transitam e se relacionam em fóruns, *chats* e murais de mensagens disponíveis nas páginas de apresentação dos *mods*. Para tanto, uma breve apresentação da Steam Workshop e da Nexus Mods é necessária.

Ela é reflexo de um trabalho ainda maior, no plano de tese, que visa, a partir de uma aproximação etnográfica em Hine (2015) e Boellstorff (2009) e netnográfica em Kozinets (2015), a investigação e consequente análise das atividades de produção, das práticas, das formas de interação e das relações exteriores, tomando suas atividades como alternativas as da indústria dos games, a partir das comunidades de *mods*.

Steam Workshop e Nexus Mods são dois ambientes que até compartilham do mesmo objetivo, contudo, suas origens não poderiam ser mais distintas. Enquanto a Nexus Mods é um coletivo independente, organizado a partir de doações, inscrições dos membros *Premium* e da publicidade gerada nas páginas da comunidade, a Steam Workshop é uma ramificação da plataforma de jogos Steam, por sua vez, pertencente à Valve, uma das principais *publishers* de jogos da atualidade.

Tanto a Steam Workshop quanto a Nexus Mods disponibilizam milhares de jogos, organizados em páginas independentes, que por sua vez são constituídas de abas ou links específicos para os conteúdos disponibilizados pelos jogadores. Para fins de pesquisa, acabamos por escolher a série de jogos de RPG da Bethesda, The Elder Scrolls, mais precisamente o quinto título da saga, Skyrim. Entre os motivos creditados, destacamos a popularidade da franquia, ainda hoje, cinco anos após o lançamento de Skyrim, um

dos jogos com mais *mods* produzidos; consequentemente, tanto na Steam Workshop quanto na Nexus Mods, podemos encontrar páginas atualizadas com *mods*, notícias, discussões, posts com imagens, vídeos etc.

Tanto na Workshop quanto na Nexus Mods de Skyrim, os conteúdos são diversos, não se limitando apenas aos *mods*. Existem guias, tutoriais, vídeos, imagens, kits de edição e manipulação. Tratando especificamente dos *mods*, percebemos na sua diversidade *mods* de correção, de armas, armaduras, personagens, *quests* (de missões alternativas), animais e monstros, aprimoramento gráfico, entre outros.

A partir da variedade das ofertas, focamo-nos com mais pormenor em três *mods* que atestam para o princípio de complexidade do universo da *fan culture*. São *mods* de *quests*, que colocam o jogador em situações narrativas alternativas daquela do jogo tradicional. Em certa medida podemos falar que estes *mods* são uma forma de *fanfic* – as histórias criadas por fãs – ainda mais imersivas que as tradicionais, uma vez que a experiência dos *games* carrega, por si só, este princípio.

Helgen Reborn, Agent of Righteous Might e Falskaar são *mods* em que é possível compreender o que Keen (2016) considera como condição hierárquica da *fanfic*. Para o autor, existe um princípio natural de conformidade em que as histórias criadas pelos fãs se situam próximo do cânone, do universo reconhecido e criado por outrem, geralmente, as grandes produtoras da indústria cultural.

Em Helgen Reborn¹, a partir da instalação do *mod*, cabe a reconstrução da vila de Helgen, destruída logo no começo do jogo por um dragão², uma das criaturas anciãs que se julgavam extintas, e que misteriosamente voltam para assombrar os céus de Skyrim. O mais interessante é que é justamente em Helgen que a aventura do jogador começa no jogo original, uma vez que foi capturada e levada a julgamento – e consequente execução – ao estar supostamente envolvida no levantamento contra o poder centralizador da província de Cyrodiil, que tem no seu artífice, o rei supremo de Skyrim, Ulfric Stormcloak.

Já em Agent of Righteous Might³, o jogador será incumbido de realizar uma série de missões que o levarão até ao seio do culto que reverencia Stendarr, a divindade da justiça. Na verdade, uma facção deste culto. Se no cânone de Skyrim temos os Vigilants of Stendarr como representantes da fé militante da justiça em nome dos 9, em *Agent of Righteous Might* temos um grupo de mesmo nome, liderados por

¹ Mais detalhes em: <http://www.nexusmods.com/skyrim/mods/35841/>

² Um dos motivos que justificam este *mod*, se dá pelo retorno do lendário dragão Alduin, uma das entidades mais poderosas de Nirn, o plano terreno ou mortal no qual se situa o continente de Tamriel e consequentemente Skyrim. Alduin é o principal antagonista na história principal do jogo (deixando em segundo plano a guerra civil proporcionada pelo levante de Ulfric Stormcloak contra os Imperials) e sua presença se dá ainda no começo do jogo, quando ele é um dos artífices da destruição de Helgen.

³ Mais detalhes em: <http://www.nexusmods.com/skyrim/mods/33766/>

Nirtius Pontaniam, antigo membro da ordem dos Vigilants. O jogador será reconhecido como um dos discípulos da ordem dos Vigilants e a partir disso, convidado para fazer parte dos Agents, recebendo missões que visam a aniquilação dos membros restantes da lendária Mythic Dawn, estabelecidos no cânone de Elder Scrolls como uma ordem herética, um culto de origem Daedrica⁴ que segue os ensinamentos de Mehrunes Dagon, um dos dezessete príncipes do mundo alternativo de Oblivion. Os sobreviventes da Mythic Dawn estão a reorganizar-se, prestes a desferir um golpe fatal no plano terreno de Nirn e cabe ao jogador, como parte dos Agents, obliterar esta ameaça.

Falskaar⁵, por fim, parte de uma misteriosa localização em Skyrim que vem produzindo de forma contínua, por mais de 600 anos, um estranho barulho. O jogador então decide investigar a origem do mistério, e nos subterrâneos de Skyrim, descobre uma passagem secreta com um portal que liga toda Tamriel a Falskaar. Depois de ajudar um dos guardas do portal com um problema referente a uma rebelião em curso, o jogador recebe permissão para explorar Falskaar e tudo aquilo que ela oferece. E é aí que tudo começa e a semelhança com Skyrim volta mais uma vez. Em Falskaar, também percebemos o conflito de duas facções rivais, neste caso, duas famílias de linhagem nobre, tradicionais e que lutam a partir de ideais distintos. O jogador, mais uma vez, se vê no meio de um conflito com o qual ele não tem relação, mas que se envolve conforme progride nas missões principais e nas alternativas. As missões envolvem batalhas, libertação de reféns, aniquilação de monstros e até compra de presentes.

A breve apresentação dos três *mods*, se não parte de uma abordagem exaustiva, ao menos nos possibilita identificar os princípios pelos quais a apropriação do cânone de Skyrim ocorre. Trata-se de um princípio que opera dentro dos limites estabelecidos do que convencionamos chamar de multiverso cultural, na qual as formas de produção cultural e alternativa da fan culture se mesclam e ampliam os horizontes das formas de manifestação cultural, situadas num espaço de intersecção e que são tomadas de roldão pelas possibilidades de protagonismo dos utilizadores vulgares.

Para já, convém tratar deste espaço, esta intersecção entre duas realidades, dois universos. Nosso próximo passo ocorre da tentativa de identificar não só este espaço, mas também esta relação que se estabelece entre dois (ou mais) universos a partir do conceito de multiverso cultural.

O MULTIVERSO CULTURAL

O emprego do termo «multiverso cultural» tem por objetivo preencher algumas lacunas no reconhecimento e sistematização de um universo de produção cultural alternativo no qual seus participantes ingressam graças às dinâmicas e rituais de sociabilização próprios.

O termo remete a ideia popularmente reconhecida dos domínios da banda desenhada da DC e Marvel, com uma leve variação nos conceitos das duas. Para a DC⁶, trata dos vários universos pertencentes a um mesmo domínio, um mesmo espaço físico compartilhado, mas que são separados de alguma maneira por ressonâncias vibracionais. (Cabe como parêntese destacar que o sitio oficial da DC disponibilizou, no ano de 2014, um mapa atualizável identificando e explicando cada um dos universos do seu multiverso; este pode ser acessado no link que segue: <http://www.dccomics.com/blog/2014/08/18/the-map-of-the-multiverse>). Já a Marvel⁷, adota um sistema hierárquico no qual os múltiplos universos formam parte de uma coleção e mantem entre eles algum tipo de ligação, ainda que preservem certo sentido de autonomia.

Ora bem, o que pretendemos ao nos apropriarmos do termo é sistematizar parte do organismo cultural, identificando universos de produção cultural e interação social distintos, inter-relacionados e relativamente autônomos e as «ressonâncias» como pontos de intersecção entre estes universos, os momentos em que desta relação, ocorrem as trocas de conteúdos, ferramentas, conhecimento, e que se produzem sentidos que operam, em certa medida, numa relação de interdependência.

Pensemos esta condição, por exemplo, quando tratamos da relação que se estabelece nos universos distintos da pirataria e da indústria cultural. Pois o seu ponto de intersecção ocorre a partir do momento em que os utilizadores se apropriam dos conteúdos culturais produzidos pela indústria cultural e os transportam para os limites do universo de disseminação alternativo da pirataria. O mesmo poderia ser aplicado se nos valêssemos de dois outros universos culturais, como os da cultura popular e o da cultura erudita em que o ponto de intersecção é tanto o consumo cultural realizado por um dos lados como a incorporação dos elementos de um universo pelo outro a partir da ação dos seus produtores, sejam eles os produtores tradicionais ou os novos agentes deste processo, os utilizadores vulgares da internet.

O multiverso cultural então sinaliza para as três características básicas que formam a ideia de cultura de convergência segundo Jenkins (2006b): a convergência midiática pela afirmação dos conteúdos e dispositivos transmedia, a inteligência coletiva e a cultura participativa.

⁴ Os Daedras – Daedric em inglês – é uma raça de poderes sobrenaturais que habitam o mundo alternativo de Oblivion.

⁵ Ver mais em: <http://www.nexusmods.com/skyrim/mods/37994/>

⁶ A ideia de multiverso segundo a DC é mais bem detalhado na página Wikia que segue em linha: <http://dc.wikia.com/wiki/Multiverse>

⁷ A ideia de multiverso segundo a Marvel é mais bem detalhado na página Wikia que segue em linha: <http://marvel.wikia.com/wiki/Multiverse>

O que temos então, é um fenómeno único no qual o fluxo dos conteúdos se desloca para o domínio do transmedia; em que o impacto central dos consumidores, agora agentes ativos na cadeia de produção e disseminação de conteúdos - uma vez transposta a barreira de apropriação dos meios de produção e dos canais de distribuição - transita em via dupla, ou seja, remodela e constrói novos sentidos, produtos, organiza e gerencia novos ambientes de discussão e interação, produção e disseminação.

Neste sentido, poderíamos assumir a *fan culture* como um universo de produção cultural à parte, na qual a soma dos seus membros, organizados em grupos particulares de filiação e identificação formam grupos e comunidades distintos. Neste contexto, e estando relacionado com os vários universos pertencentes do multiverso cultural, a *fan culture* acaba por ter a sua disposição tanto os dispositivos de produção e distribuição de conteúdos quanto dos próprios conteúdos, uma vez que sua relação se expressa manifestamente a partir das produções culturais tidas como tradicionais. O universo da *fan culture* acaba por ser o universo ideal em termos da apropriação dos meios de produção, dos canais de distribuição e, naturalmente, dos conteúdos.

É neste momento que se torna necessário recuperar Almeida (2009) que ao tomar uma perspectiva que aproxima a mediação ao consumo e a crítica cultural, sugere que «as tecnologias da informação e comunicação (TICs), ao descortinarem inéditas formas de produção, circulação e recepção de produtos simbólicos, também contribuíram para tornar a cena cultural contemporânea cada vez mais complexa». O autor prossegue ao afirmar que «múltiplas camadas de informação se agregam aos produtos culturais e em torno deles, sinalizando a constituição de um novo tipo de 'conhecimento' necessário para a crítica, o consumo, e, no limite, para a própria compreensão dessas mercadorias singulares». A internet, neste sentido, seria ela mesma um mecanismo de mediação e por consequência, um ambiente favorável às novas formas de mediação, ou assumindo a posição de Silva e Lopes⁸ (2011), um universo convidativo as formas de desintermediação levadas a cabo de forma zelosa por seus utilizadores⁹, os novos agentes de distribuição e compartilhamento.

A função desintermediadora é, então, parte fundamental na expressão da *fan culture*, tornando mais nítida a

compreensão dos processos de troca que ocorrem tanto no seu universo quanto na relação com os demais universos do multiverso cultural. Os consumidores, deslocados da sua função passiva, transitam agora num ambiente em que são também reconhecidos como mediadores, daí a condição desintermediadora da *fan culture* na qual nos referimos.

A estrutura de camadas do multiverso cultural

Ora, apresentamos os *mods*, inseridos no contexto de comunidades estabelecidas no universo da *fan culture*; na sequência introduzimos o modelo de multiverso cultural, mas afinal, como podemos relacionar estas duas realidades? O caminho mais promissor nos parece ser o de introduzir uma análise a partir das camadas existentes no multiverso cultural, nomeadamente as camadas de apropriação que, acreditamos, traduzem o sentido desta relação.

Estas camadas de apropriação podem ser entendidas como processos de validação dos pontos de intersecção entre dois universos distintos, que aqui são representados pelo universo da *fan culture* e o universo canónico estabelecido pela Bethesda, a produtora de The Elder Scrolls Skyrim.

A ampliação da estrutura narrativa (presente nos três *mods*), o incremento dos mapas com novas localidades e personagens para se interagir (em Falskaar), a incorporação, a partir da influência canónica, de novos elementos que se misturam a mitologia da série (como em Agent of Righteous Might), o implemento de novas funcionalidades e ferramentas (como em Helgen Reborn), são, por exemplo, expressões manifestas das camadas de apropriação das quais os produtores se utilizam para expandir a experiência do jogo.

Neste sentido, estas adaptações constituem verdadeiros recortes, mixagens a partir do cânone de The Elder Scrolls no qual os jogadores-produtores são estimulados a esmiuçar, reconstruir sentidos e produzir de forma aditiva estes novos conteúdos. É por associação o princípio no qual opera a ideia de cultura R/W de Lessig (2008), do universo dos *remixes*, das trocas que são estabelecidas numa via de mão dupla de produção de conteúdos.

E se a condição de apropriação não está expressa na camada canónica, pode também estar vinculada a camada de construção narrativa, na qual os criadores dos *mods* acrescentam formas de manipulação naquilo que Campbell identifica como a «jornada do herói». Também conhecida como monomito, este conceito se refere aos mais variados elementos que as narrativas mitológicas partilham entre si, independente da sua procedência ou do tempo que foram concebidas. Posteriormente, Christopher Vogler¹⁰ partiu das teorias de Campbell para formular uma espécie de guia em que propunha uma espécie de estruturação narrativa para

⁸ Segundo as autoras: «O surgimento da Internet transformou a rede em infinita encruzilhada, que subverte a noção de espaço local e global, que rompe com sistemas de intermediação tradicional entre emissores e receptores ao permitir que cada nó da rede possa produzir e distribuir mensagens. A distribuição de informação na tipologia de rede anterior caracteriza-se como de um para todos, e agora essa distribuição passa a ser de todos para todos» (Silva, Lopes, 2011).

⁹ Para Almeida (2014) o que vemos então é um processo de descentralização no qual as formas institucionalmente aceites de mediação cedem frente ao apelo dos usuários enquanto seus próprios mediadores. Logo, «a internet solidificaria sua autonomia frente as suas necessidades informacionais e culturais» (Almeida, 2014, p. 199).

¹⁰ O livro em questão é o « The Writer's Journey: Mythic Structure for Storytellers and Screenwriters, », publicado no ano de 1992.

escritores e roteiristas. Em ambas podemos identificar três passos importantes: a partida, a iniciação e o retorno.

Tanto em Helgen Reborn, Agent of Righteous Might e Falskaar podemos perceber estas três etapas, até mesmo partes que constituem as 17 etapas do ciclo de três fases de Campbell, como, por exemplo, o «chamado da aventura», quando as narrativas guiam os jogadores para o início da jornada, na apresentação de um problema/mistério a ser resolvido, quebrando assim uma dada rotina; no «auxílio sobrenatural», quando, em Agent of Righteous Might, Nirtius Pontaniam nos guia para o submundo da Agents e da caça a Mythic Dawn; «a passagem pelo primeiro limiar» quando o jogador entra em um novo mundo, uma nova experiência ou realidade, a exemplo do que acontece com a passagem para Falskaar e etc.

À esta camada de transformação narrativa, temos também uma outra, talvez não tão evidente nos *mods* apresentados, mas que é parte primordial na fundamentação destes enquanto mecanismos de imersão alternativos: a transformação da «realidade material» do jogo, dos elementos que ajudam a compor a camada narrativa. Trata-se de uma camada um tanto quanto sutil, técnica, e que pode às vezes se confundir com a camada narrativa por estar subordinada a esta. Entretanto, a transformação da realidade material do jogo trata dos implementos concretos, dos objetos, construções, de novos personagens, monstros, localizações, armas e até mesmo das formas de interação a partir das batalhas, o que fica mais evidente quando buscamos dois outros mods, Warzones e Populated Skyrim Civil War Reborn¹¹.

De todo modo, o que vemos é a utilização de uma base já estabelecida como ponto de partida para o trabalho destes jogadores-produtores. Suas histórias, sejam elas a dos *mods* ou das apresentações dos seus personagens, estão situadas numa espécie de intersecção, um ambiente em que, ao menos em benefício dos jogadores, existe condição para se apropriar dos conteúdos, das ferramentas e dos canais de distribuição e incrementar o universo canônico de Skyrim. Entretanto, esta não é uma via de mão única: neste ambiente mesclado, a Bethesda, produtora da série The Elder Scrolls, passa a ter a sua disposição parte de uma vasta rede de colaboradores, (inter)dependentes da sua produção; mas que ao mesmo tempo, ao criarem e recriarem tomando o cânone como base, potencializam o alcance da série, permitem a extensão do seu universo regular e o fortalecimento da sua comunidade.

¹¹ Warzones e Populated Skyrim Civil War Reborn são dois *mods* nos quais os limites da Guerra civil que toma conta de Skyrim são testados. A proposta é que o jogador sinta os efeitos devastadores de um conflito desta magnitude ao ter em seu jogo tanto a multiplicação dos focos de conflito entre Stormcloaks e Imperials quanto da quantidade de personagens em cada um destes conflitos. Mais informações sobre Warzones em: <http://www.nexusmods.com/skyrim/mods/9494/> e Populated Skyrim Civil War: <http://www.nexusmods.com/skyrim/mods/73221/>

CONSIDERAÇÕES PARCIAIS

A apresentação dos *mods*, seguida da introdução do modelo de multiverso cultural e das camadas de apropriação que validam as relações entre os universos de produção cultural estão inseridas no âmbito de uma tese de doutoramento a ser realizada tomando como ideia a análise das atividades desenvolvidas nas comunidades de *modding* Nexus Mods e Steam Workshop.

Neste primeiro momento, julgamos ter dado um passo importante na construção de um modelo teórico auxiliar que permita situar as atividades e as formas de interação e produção a partir da dupla integração entre produtores alternativos da fan culture e a produção da indústria cultural. Não se trata, contudo, de um modelo acabado, no que nossos esforços estão concentrados para, no seguimento da pesquisa, aprimorar tanto o modelo quanto aprofundar as análises dos objetos de investigação.

O multiverso cultural é uma força de ação colectiva, relativamente independente, essencialmente cooperativa, que por muitas vezes pode acabar por produzir tensões nos seus pontos de intersecção, haja vista multiplicidade de sujeitos e organizações envolvidos, cada qual com seus interesses e projetos particulares. Contudo, ao nos focarmos nos seus aspectos mais positivos procuramos não eliminar ou minimizar estas tensões, mas indicar, a partir das suas camadas de apropriação, como os pontos de intersecção podem funcionar como um mecanismo de reverberação, aprimoramento e ampliação de universos culturais distintos e do multiverso como um todo.

Neste sentido, o universo dos videojogos acaba por nos servir de valiosa amostra pela gama de atividades criativas que ocorrem nos domínios da sua *fan culture*; seja na criação de conteúdos, ferramentas, aplicativos e até mesmo em novos ambientes de promoção, discussão e participação, estamos diante de um cenário em constante transformação, que traz tanto algumas dificuldades quanto desafios.

REFERÊNCIAS

1. Almeida, M. A. (2009). Informação, tecnologia e mediações culturais. *Perspectivas em Ciência da Informação*, 14, N.E., 184-200.
2. Almeida, M. A. (2014). Mediação e mediadores nos fluxos tecnoculturais contemporâneos. *Informação e Informação*, 19, 2, 191-241.
3. Boellstorff, T. (2009). *Coming of age in Second Life: an anthropologist explores the virtually human*. New Jersey: Princeton University Press
4. Bruns, A. (2009). From prosumer to produser: understanding user-led content creation. *Transforming Audiences*, 3, 4.

5. Cardoso, G. (2006). *Os media na sociedade em rede*. Lisboa: Fundação Calouste Gulbenkian.
6. Castells, M. (2012). *A sociedade em rede. A era da informação, economia, sociedade e cultura, 1*. São Paulo: Paz e Terra.
7. Gee, J. P. (2007). *What video games have to teach us about learning and literacy*. (2ª ed.). New York: St. Martin's Griffin
8. Gere, C. (2008). *Digital culture*. Londres: Reakiton Books.
9. Grossberg, Lawrence. (1992). Is there a fan in the house? The affective sensibility of fandom. In Lisa A. Lewis, *The adoring audience: fan culture and popular media* (pp. 50-65). New York: Routledge.
10. Hong, R., Chen, V. H-H. (2014). Becoming an ideal co-creator: Web materiality and intensive laboring practices in game modding. *New Media & Society*, 16, 2, 290-305.
11. Hills, M. (2005). *Fan cultures*. New York: Routledge.
12. Hine, C. (2015). *Ethnography for the internet: embedded, embodied and everyday*. New York: Bloomsbury.
13. Jenkins, H., Ito, M. Boyd, D. (2016). *Participatory culture in a networked era: a conversation on youth, learning, commerce, and politics*. Cambridge: Polity Press.
14. Jenkins, H. (2006a). *Fans, bloggers and gamers: exploring participatory culture*. New York: New York University Press.
15. Jenkins, Henry. (2006b). *Convergence culture: Where old and new media collide*. New York: New York University Press.
16. Jenkins, Henry. (1992). *Textual poachers: television fans & participatory culture*. New York: Routledge.
17. Kozinets, R. (2015). *Netnography: redefined*. (2ª ed.). Londres: SAGE Publications.
18. Lessig, L. (2008). *Remix: making art and commerce thrive in the hybrid economy*. New York: Bloomsbury.
19. Poor, N. (2014). Computer game modders' motivations and sense of community: a mixed-methods approach. *Games and Culture*, 16, 8, 1249-1267.
20. Raine, L., Wellman, B. (2014). *Networked: the new social operating system*. Londres: The MIT Press.
21. Ritzer, G., Dean, P., Jurgensen, N. (2012). The coming age of the prosumer. *American Behavioral Scientist*, 59, 4, 379-398.
22. Skyrim Nexus Mods. (2016). *Agent of Righteous Might*. Disponível em: <http://www.nexusmods.com/skyrim/mods/33766/>?
23. Skyrim Nexus Mods (2016). Falskaar. Disponível em: <http://www.nexusmods.com/skyrim/mods/37994/>?
24. Skyrim Nexus Mods. (2016). Helgen Reborn. Disponível em: <http://www.nexusmods.com/skyrim/mods/35841/>?
25. Sotamaa, O. (2010). When the game is not enough: motivations and practices among computer game modding culture. *Games and Culture*, 5, 3, 239-255.
26. Toffler, A. (1989). *The third wave*. New York: Bantam.
27. Van Dijck, J (2012). *The culture of connectivity: a critical history of social media*. New York: Oxford University Press.

Desenvolvimento de um Jogo para o ensino de Física com Design Centrado no Usuário

Jessica Leite Pituba

Escola Politécnica - Universidade de São Paulo
Av. Prof. Luciano Gualberto, 380 - Butantã - São Paulo, Brasil
jessicapituba@usp.br

Ricardo Nakamura

Escola Politécnica - Universidade de São Paulo
Av. Prof. Luciano Gualberto, 380 - Butantã - São Paulo, Brasil
ricardonakamura@usp.br

RESUMO

O design de jogos educativos deve agradar seu público-alvo, além de adaptar bem o conteúdo e respeitar o seu contexto social. Uma forma de avaliar se o jogo é do agrado do jogador é envolvê-lo no processo de design, que é chamado de design centrado no usuário. Neste trabalho, é apresentada uma sessão de jogo com 4 alunos que fazem parte do público alvo e 3 jogos que utilizem mecânicas de física para auxiliar com o processo de concepção do design do jogo com as respostas dos alunos sobre os jogos que experimentaram. Após isso, apresentamos o jogo desenvolvido, que é um jogo de interface simples que aborda o conteúdo de atrito.

ABSTRACT

The game design for educative games should please the target public, while adapting the educational content and respecting their social context. One way to evaluate if the game is pleasant to its public is involving the players on the design process, also called a user centered design. On this work, it is presented a game session with 4 students that are part of the target public and 3 games that utilize physics in its mechanics, to help with information for the design conception through the answers of the students about the games they played. At the end, the game developed is presented and it is a game with simple interface that covers the content of friction.

Palavras-Chave

design de jogos; design centrado no usuário; game design; user centered design; jogos educativos; serious games;

INTRODUÇÃO

Segundo Prensky [12], o design de jogos educativos deve levar em conta 3 aspectos: O público alvo, em seus gostos e outros jogos que já conheçam, contexto social e conteúdo. Desenhar um jogo educativo com características já aprovadas pelos alunos, aumenta a chance de adesão àquele recurso de aprendizado. Entretanto, quando o público-alvo do jogo se torna muito abrangente, como por exemplo: “qualquer aluno do primeiro ano do superior de engenharia” ao invés de uma única classe, é difícil isolar um único estilo de jogo preferido e compatível com o conteúdo a ser apresentado. Nesse contexto, uma abordagem possível é a de jogos

adaptativos [10], nos quais a dificuldade ou mesmo as mecânicas são ajustadas conforme o perfil do jogador. Para esses casos, pode-se utilizar outros recursos projetuais do campo do Design. Um deles é o design centrado no usuário, que propõe desenvolver soluções centralizadas no usuário e em suas necessidades.

A ideia de centralizar o design no usuário é similar ao objetivo do design de games de criar uma experiência para o jogador. Utilizando as ferramentas do design centrado no usuário com as técnicas de design de games, temos o chamado Design Centrado no Jogador (Player-Centered Design).

Neste artigo, discutimos o uso de técnicas de design centrado no usuário para o projeto de jogos educacionais adaptativos. Para isso, apresentamos uma atividade do processo de design de um jogo de Física para alunos de engenharia: um experimento de design centrado no usuário realizado com 4 alunos do público-alvo do jogo em uma sessão interativa com jogos que utilizam conceitos físicos como suas mecânicas de jogo. Apesar de nem todos os jogos utilizados possuírem um objetivo educacional, foi possível realizar uma análise dos pontos fortes e fracos de cada jogo, segundo os jogadores. Em seguida, há uma discussão sobre o impacto das descobertas desse experimento para o desenvolvimento desse jogo.

REVISÃO BIBLIOGRÁFICA

Esta seção trata de alguns conceitos utilizados para essa pesquisa: Design centrado no usuário, *meaningful play*, canal de *flow* e desenvolvimento de jogos educacionais.

Design Centrado no Usuário

O termo “Design Centrado no Usuário” é utilizado para cunhar vários métodos cujo objetivo é envolver o usuário final no processo de design. Esse termo foi criado por Norman em seu laboratório de pesquisa em 1980, mas se popularizou apenas com a publicação de um livro com Stephen Draper, Design de Sistemas Centrados no usuário: Novas perspectivas em Interação Humano-Computador (1986). Norman desenvolveu o conceito em seu livro O Design do Dia-a-Dia (1988) [9], onde ele trata com maior profundidade a usabilidade do design,

reconhecendo as necessidades e interesses do usuário. Nessa obra, também propõe 7 princípios para o design centrado no usuário:

1. Usar, ao mesmo tempo, o conhecimento no mundo e o conhecimento na cabeça.
2. Simplificar a estrutura das tarefas
3. Tornar as coisas visíveis: assegurar que as lacunas de execução e avaliação sejam encurtadas ou superadas
4. Fazer corretamente os mapeamentos
5. Explorar o poder das coerções naturais e artificiais
6. Projetar para o erro
7. Quanto tudo mais falhar, padronizar

Norman destacou a necessidade de trazer usuários reais do produto para participar do design. Atualmente, isso é feito de várias formas e em diferentes pontos dos projetos de design. O usuário pode ser entrevistado sobre suas preferências, observado e até mesmo participar de uma simulação com protótipos de papel do produto final. Há também o teste de usabilidade, onde o usuário é envolvido quando o produto está em sua fase alfa ou beta, para detectar possíveis falhas no design de uso.

Outra forma de envolver o usuário no processo é o design participativo, onde um grupo de usuários do público alvo participa ativamente do design, tornando-se co-designers. Nesse tipo de processo, os designers agem como facilitadores, explicando a parte mais técnica do design de produtos e corrigindo eventuais falhas, enquanto os usuários realizam o processo de criação de acordo com o que lhes parece melhor.

Envolver os usuários no design pode ser um pouco mais custoso, mas facilita para que o designer entenda as necessidades e expectativas do usuário sobre o novo produto. Essas características são fundamentais para um design com boa usabilidade.[1]

Meaningful Play e Canal de Flow

Meaningful Play é um conceito que trata de criar significado através da experiência do jogar. Segundo Salen e Zimmerman [15], ele é apresentado como um retorno que o jogo fornece ao jogador após uma ação ser executada. Este retorno pode ser a curto ou a longo prazo.

A curto prazo, esse retorno apresenta-se como uma representação da situação atual do jogo, logo após uma ação do jogador. Um jogo de tabuleiro, por exemplo, precisa deixar claro com suas peças e o posicionamento delas qual foi a mudança causada pela ação do jogador naquele turno. Similar a um tabuleiro, o jogo eletrônico também precisa demonstrar ao jogador o resultado de suas ações de forma simples, que frequentemente é uma alteração visual no cenário do jogo.

O significado a longo prazo é o significado criado pelo progresso dentro do jogo. Um exemplo simples desse significado é o ganho de um nível após um certo acúmulo de experiência, que pode trazer consigo novas habilidades para o jogador utilizar, ou apenas demonstrar que o jogador está avançando mais no jogo e que seus desafios se tornarão mais difíceis. O significado ao longo prazo é importante, pois motiva o jogador a continuar experimentando durante o jogo. Em seu livro, Raph Koster [6] diz que jogadores não irão realizar tarefas que não lhes dá nenhuma recompensa no jogo, ou que não consigam perceber que há uma recompensa por aquilo. O jogador não vê sentido em se esforçar por algo que não irá produzir resultados para a sua experiência.

Idealmente, o *meaningful play* seria uma forma de criar uma experiência única para cada jogador do jogo, com as suas ações guiando o jogo para um caminho diferente de outros jogadores.

O Canal de *Flow* é outro conceito ligado a experiência de jogo, porém, esse conceito trata da concentração e diversão que o jogador tem durante o jogo. O conceito vem da teoria do psicólogo húngaro Mihalyi Csikszentmihalyi sobre o nível de concentração criado por um indivíduo durante uma atividade que este considera prazerosa. Dentro desse estado, o indivíduo perde a percepção do mundo externo e do tempo passado.

Essa concentração pode ser alcançada em qualquer tipo de tarefa, desde que ela cumpra com 4 requisitos estabelecidos pelo psicólogo e explicados no livro de Jesse Schell [13]: objetivos claros, nenhuma distração durante a tarefa, feedback direto e desafio contínuo e equilibrado. É preciso que os objetivos estejam bem claros durante a tarefa, para que o indivíduo possa se concentrar apenas em alcançar esses objetivos. Do contrário, ele irá se preocupar demais em descobrir o que deve fazer e não irá apreciar a realização da tarefa. Não podem haver distrações dentro do ambiente onde a tarefa será realizada, para que a concentração seja criada e mantida durante a tarefa. Conforme o nível de concentração aumenta, há uma tolerância maior para distrações externas, porém, a distração nunca pode vir da própria tarefa, ou, além de quebrar a concentração, o indivíduo pode perder o interesse pela tarefa. O feedback direto é necessário pelo mesmo motivo que os objetivos claros. Se o ator da tarefa se preocupar mais em descobrir qual o efeito de suas ações do que realizá-las, a concentração não conseguirá se estabelecer. Por fim, o mais importante de todos para que se mantenha a concentração, é o desafio contínuo e equilibrado. Para que a tarefa seja, de fato, prazerosa, o desafio precisa corresponder as habilidades do ator da tarefa, do contrário, se tornará uma tarefa chata ou frustrante. O desafio oferecido deve ser projetado para o nível de

habilidade esperado para o público-alvo daquela tarefa e aumentar constantemente em uma função esperada após a realização de cada pedaço da tarefa. É difícil fazer um desafio justo para um público com nível de habilidades variável. O ideal é contemplar a maior parte do público dentro do desafio da tarefa. Em jogos, isso é solucionado de duas formas: níveis de dificuldade diferente de jogo, que já abrange boa parte do público, mas falha em satisfazer as pessoas que se encontram entre esses dois níveis, e o conteúdo extra junto a tolerância a falhas, que torna um nível um pouco mais difícil ou mais fácil, dependendo de como o jogador se sai e satisfaz com a tarefa completada.

Ainda assim, é difícil manter um desafio contínuo, por isso há pesquisa de novas formas de desenhar desafios e também como personalizar esse desafio para o jogador durante a sessão de jogo.

Design de Jogos Educacionais

Prensky, em seu livro *Aprendizagem Baseada em Jogos* [12], define a aprendizagem utilizando jogos como um equilíbrio entre a aprendizagem e a motivação. Ele cria um eixo com 4 quadrantes, onde variam a motivação e a aprendizagem. Os jogos tradicionais são mostrados como grande motivação e nenhuma aprendizagem e a aprendizagem virtual como um ambiente com pouca motivação e pouca aprendizagem (segundo ele, estes são obsoletos e ineficientes). Como definido acima, os jogos educacionais estão no quadrante de alta aprendizagem e alta motivação e o quadrante de alta aprendizagem com baixa motivação está vazio, pois seria impossível ter muita aprendizagem sem que o aluno esteja motivado.

A aprendizagem tradicional, assim como a aprendizagem baseada em computador, já utiliza vários elementos encontrados nos jogos, como o estímulo a tentar e errar sem consequências irreversíveis, o desafio crescente apoiando a construção de conhecimento e o aprendizado por experimentação. O aprendizado virtual ainda se aproveita das premiações, comuns dos jogos, para incentivar o aluno a continuar aprendendo. Tornar-lo um jogo seria, portanto, incluir as mecânicas de jogos ainda não incluídas pelo aprendizado tradicional.

Prensky sugere que o design de jogos educacionais, ou o uso de um jogo para aprendizagem, deve se basear na atividade do conteúdo que será abordado com o jogo. Isso significa que jogos para diferentes conteúdos devem ter como foco diferentes atividades, como memorização, exploração, experimentação, associação, entre outras. Através da atividade, pode-se escolher um gênero de jogo compatível para utilizar. Além do conteúdo, precisa-se considerar também os gostos do público-alvo que irá utilizar o jogo para a aprendizagem e também o seu

contexto social, para que o jogo não seja ofensivo ou imoral dentro daquela cultura.

Apesar desses itens que auxiliam em boa parte do design do jogo, desenvolver um jogo educativo que equilibre a aprendizagem e a motivação ainda é uma tarefa difícil. Para Prensky, quando em dúvida, deve-se priorizar a diversão, pois se a motivação for prejudicada, junto com ela será a aprendizagem, mas o inverso é mais difícil.

DESIGN DE GAMES COM DESIGN CENTRADO NO USUÁRIO

Design de games, segundo Salen e Zimmerman [15], é o processo pelo qual o designer cria uma experiência para um jogador. Nesta definição, o usuário já está colocado como centro do design de games e, considerando-se os princípios de Norman junto às dimensões apresentadas no livro *Regras do Jogo*, podemos encontrar várias correlações entre as ideias dos dois. Alguns exemplos são a recomendação de tornar informações do jogo visíveis, essencial para que o jogador entenda o jogo e também para o *Meaningful Play*; projetar para o erro, pois o jogador deve ter a liberdade de errar sem que isso o impeça completamente de prosseguir; e utilizar o conhecimento já existente ou comum, onde o jogo se aproveita das coisas que o jogador já está familiarizado para criar um novo mundo fantasioso, porém com sentido.

A quantidade de envolvimento do usuário durante o processo de design varia de processo para processo, mas, considerando os processos sugeridos por Salen e Zimmerman [15], Chris Crawford [3] e Jesse Schell [13], o usuário só é incluído após existir, pelo menos, um protótipo do jogo. Salen e Zimmerman [15] sugerem uma participação ativa dos jogadores com os protótipos em um desenvolvimento iterativo, enquanto Crawford [3] apenas inclui o usuário quando o jogo está em fase beta. Jesse Schell [13] descreve o ciclo de design como um Designer pensando em uma experiência, criando um protótipo que proporcione essa experiência para o jogador e então testando esse protótipo junto com seu jogador para avaliar essa experiência. Porém, ao desenhar a experiência, o designer não sabe exatamente quais são as expectativas e preferências do jogador a quem ela se destina. O envolvimento do jogador nessas fases iniciais poderia trazer mais certezas sobre como atingir o público alvo do jogo da maneira que se espera.

No trabalho de Duh et al [5], foi usado um método narrativo de design participativo para desenvolver um jogo para crianças de 10 a 15 anos. O objetivo era criar

um jogo educativo com uma narrativa que chamasse a atenção daquela faixa etária. Os resultados foram positivos e com pouco custo extra. Jogadores novos gostaram da forma como o jogo foi feito, enquanto os jogadores que participaram do design se sentiram importantes por poder influenciar em como o jogo foi feito. Em outros tipos de jogos, com um público alvo mais abrangente, o design participatório pode não ser viável, mas é possível usar outras técnicas comuns do design centrado no usuário para envolver o jogador, como entrevistas, observações e até simulações de ideias similares a que se quer construir.

Neste trabalho, o jogo a ser desenvolvido possui uma proposta educativa e o objetivo do envolvimento do jogador em fases iniciais do desenvolvimento é adaptar o estilo a algo mais prazeroso para os jogadores e evitar características encontradas em outros jogos do tipo que possa frustrá-los e desmotivá-los a continuar jogando.

EXPERIMENTO DE DESIGN CENTRADO NO USUÁRIO

O experimento de design centrado no usuário foi realizado através de uma sessão de jogo com 3 jogos e 5 minutos máximos para cada um dos jogos, totalizando em uma sessão de 15 minutos de jogo, e com uma entrevista qualitativa após a sessão para que o jogador expressasse sua opinião de forma explícita. Nas subseções serão explicados detalhes de análise, execução e resultados do experimento.

Método de Análise

Os métodos escolhidos para a análise do experimento foi a análise de fala para a entrevista qualitativa, para identificar coisas que o jogador não quis ou não soube explicitar em sua fala, e a análise de imagens em movimento, pois o experimento foi gravado [2]. Contudo, a análise de imagens em movimento será simplificada a gestos voluntários, pois os participantes estavam sentados e, durante uma sessão de jogo, é comum não ter muito movimento. Além desses dois métodos, também foi considerado o desempenho no jogo, como pontos de dificuldade comum e pontos que geraram mais dúvidas para os usuários durante o experimento.

Jogos Utilizados

Os jogos utilizados para o experimento foram 3 jogos que possuíam física como mecânica de jogo, focando no conteúdo das primeiras leis de Newton, pois esse é o conteúdo que se desejava abordar no jogo a ser desenvolvido. Segundo a definição, as mecânicas de um jogo são os métodos invocados por agentes para interagir com o mundo onde o jogo se passa [14], ou seja, os jogos precisam utilizar-se de regras da física como uma forma de interação, não apenas apresentar a física em sua simulação de mundo.

Como o público do jogo foram alunos do curso de engenharia, escolhemos jogos que utilizavam os conceitos das leis de Newton, que deve ser bem conhecido pelos usuários. Os jogos selecionados foram:

- Mecanika [7], ilustrado na Figura 1, um jogo do gênero puzzle desenvolvido para o ensino de física pela CREO inc. junto com professores da Faculté des sciences de l'éducation (UQAM).
- No jogo, o objetivo do jogador é levar os robôs nomeados como “Scouts” através do cenário para coletar estrelas, que devem ser tocadas por pelo menos 2 robôs para a coleta. Esses robôs não possuem funções motoras, portanto, devem ser movidos através de outros robôs que criam campos gravitacionais direcionais, impulsos ou movimentos giratórios.
- Drawtopia [4], ilustrado na Figura 2, é um jogo de gênero puzzle para celular, que apresenta um cenário simples com o objetivo de levar uma bola até uma caixa que seria o ponto final. A trilha que leva a bola deve ser desenhada pelo jogador e dá a liberdade para que o jogador solucione o nível do seu jogo com a quantidade de tinta disponível. Há também objetos coletáveis em cada nível, em forma de estrelas, que desbloqueiam outras fases mais adiante no jogo.
- Newton's Law [8], ilustrado na Figura 3, é um jogo de desenvolvimento independente do gênero ação com elementos de tiro. O ambiente do jogo não possui gravidade e o jogador necessita utilizar a sua pistola para navegar, dando o impulso para o lado contrário que deseja ir, além de combater robôs inimigos e resgatar reféns pelo cenário. Dentro do jogo também há as mecânicas de aprimoramento das habilidades do personagem, melhorando o alcance do seu tiro, a potência da arma e a velocidade.

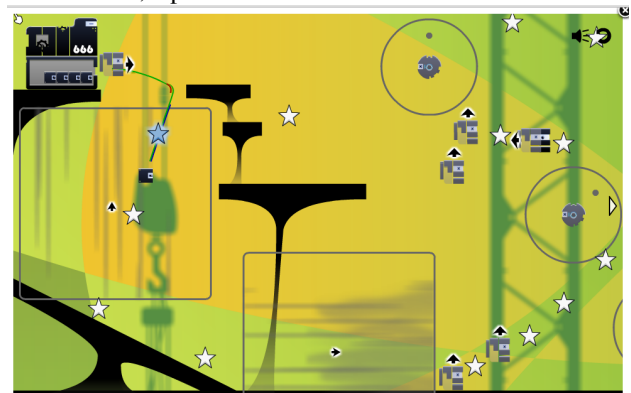


Figura 1. Jogo de puzzle, Mecanika.

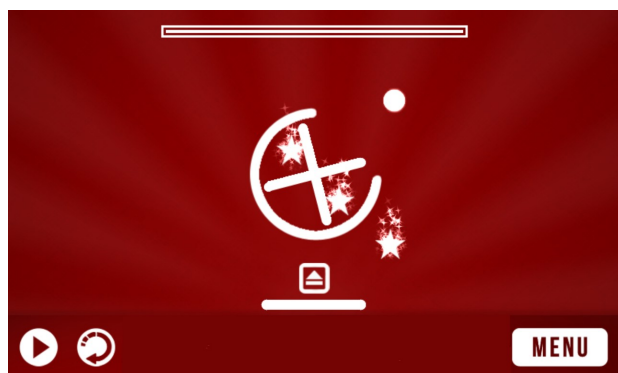


Figura 2. Jogo de puzzle para celular, Drawtopia.

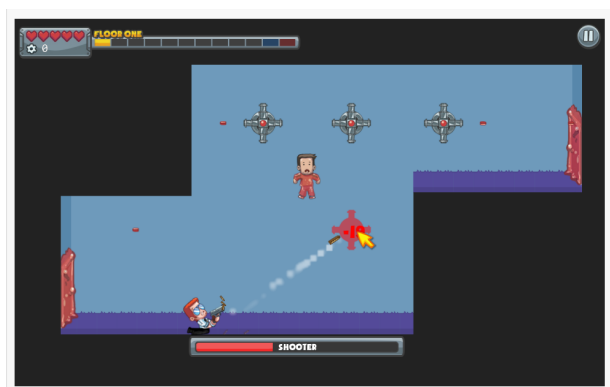


Figura 3. Jogo indie de ação, Newton's Law.

Participantes

Os participantes do experimento foram alunos voluntários da Escola Politécnica da Universidade de São Paulo que cursavam o 2º ano do curso de Engenharia. Na Escola Politécnica, os 2 primeiros anos dos cursos de Engenharia são unidos, chamado de biênio, onde são apresentados conceitos comuns para todos os cursos, tais como Física. Portanto, os alunos já possuíam os conhecimentos que foram tratados nos jogos escolhidos, o que os distancia um pouco do público-alvo, alunos do 1º ano, que não teria esse conhecimento ainda, mas auxilia na avaliação do que pode ou não ter sido deixado claro quanto ao conhecimento científico dentro dos jogos.

5 alunos se voluntariaram para participar do experimento, porém o resultado de um deles foi descartado por falha durante as gravações, de forma que apenas 4 deles foram levados em consideração durante a análise dos resultados.

Protocolo

O estudo foi conduzido nas dependências da própria Escola Politécnica, em um lugar onde estavam presentes somente o aluno e o avaliador, com o equipamento de montagem montado para capturar as imagens.

O jogador teve uma sessão de jogo de 15 minutos, sendo 5 minutos distribuídos para cada um dos jogos

selecionados para o experimento. A ordem era aleatória e foi permitido que o jogador trocasse de jogo no momento que quisesse. Após a sessão, foi realizada uma curta entrevista com 4 perguntas para estimular o participante a expressar sua opinião de forma mais explícita. As perguntas eram:

1. O que você achou dos jogos?
2. O que você mais gostou em cada um deles?
3. E o que menos gostou?
4. Já havia jogado algum dos jogos apresentados neste experimento ou similar?

Resultados

Durante as entrevistas, nenhum jogador apontou claramente um dos jogos como o mais fácil ou mais difícil. As opiniões sobre cada um dos jogos divergiram, o que estava de acordo com a experiência diferente que cada aluno teve com cada jogo, mas houveram pontos em comum criticados e elogiados para cada um deles.

Em Mecânica, a opinião geral foi que o jogo trazia uma mecânica muito interessante utilizando os campos de gravidade direcionais. Um dos jogadores afirmou que o mais interessante do jogo é que parecia mais próximo da realidade, o que pode ter ajudado na hora de relacionar a física do mundo do jogo com mundo real. Entretanto, o jogo era de difícil entendimento a primeira vista. Ao começar uma fase, um gesto comum era o de colocar a mão no queixo, enquanto lia as instruções escritas ou olhava a disposição dos robôs na tela. A barreira da língua pode ter influenciado nessa dificuldade de aprendizado, pois as instruções apresentadas na tela estavam na língua inglesa, porém acredita-se que a maneira de explicar o que deveria ser feito não estava clara o suficiente. Um exemplo é a primeira fase, onde o lugar para o robô que cria campos direcionais é apontado, junto ao gesto de arrasta-lo, ilustrado na Figura 4. a). Os jogadores entendem que devem arrasta-lo, mas não entendem a especificidade do local até ativarem o robô de fabricação dos Scouts, os robôs sem movimento, para coletarem as estrelas, situação ilustrada na Figura 4. b). Apenas após os Scouts sumirem, o jogador percebe que algo está errado e volta a interagir com a tela. Um ponto positivo é que essa dificuldade só foi observada até, no máximo, o segundo nível. Depois disso, o jogador sente-se mais confortável para experimentar e alterar a disposição dos robôs quando não consegue o resultado esperado.

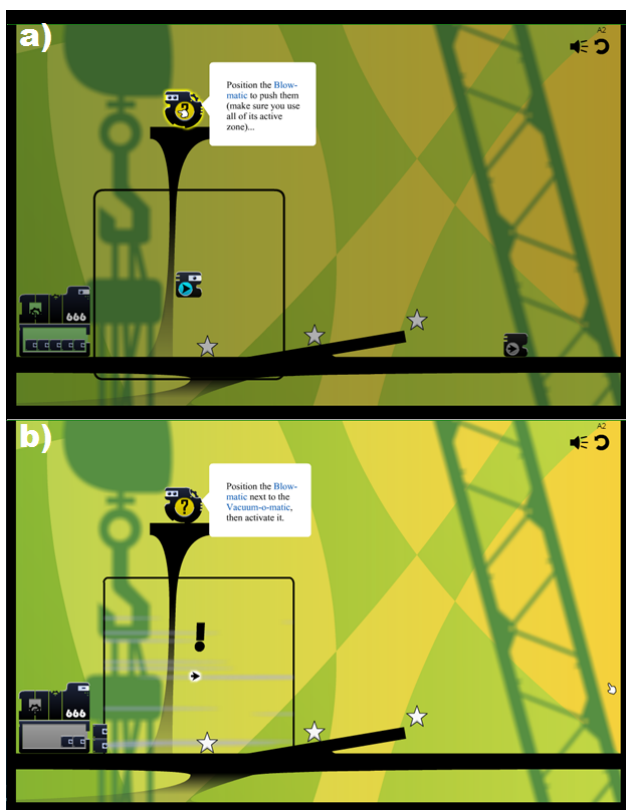


Figura 4. a) tutorial explicativo do jogo no primeiro nível, arrastando o robô. b) erro comumente cometido pelos jogadores do experimento após ver o tutorial.

Drawtopia foi um jogo que ficou neutro para a opinião dos jogadores. Ele se apresentou como um jogo de gráficos simples e minimalistas, de forma que não houve dificuldades para entender do que se tratava o objetivo do jogo. Porém, os alunos acharam incômodo a precisão do jogo ao desenhar o trajeto da bola, tornando o jogo mais difícil do que proposto a princípio. Nas filmagens, isso pode ser observado nos níveis de aprendizado, onde o jogador não possui dificuldades em traçar a linha seguindo o guia na tela, mas, mesmo com a solução certa, sua bola acaba parada em algum lugar ou não atinge a caixa onde o nível termina. Ao remover as ajudas pontilhadas, o gesto de colocar a mão no queixo retorna após uma primeira tentativa de resolver a fase. Junto a ele, outro gesto, balançar de cabeça, logo antes de uma tentativa quase certa. A precisão do jogo dava a falsa informação de que o desenho estava errado e levava o jogador a tentar coisas diferentes que também não funcionavam, até voltar ao resultado inicial que, quando desenhado com mais cuidado, chegava ao objetivo.



Figura 5. a)Exemplo de nível com robôs passivos. b) Exemplo de nível com robôs agressivos

Por fim, Newton's Law empolgou os jogadores com uma mecânica diferente e uma história levemente satírica, porém frustrou na dificuldade desbalanceada. Os jogadores começavam o jogo experimentando a navegação na sala inicial, onde o personagem já se encontrava em um ambiente sem gravidade, antes de seguir pela porta para o primeiro nível onde poderiam surgir inimigos. Os inimigos dos jogos eram todos robôs, havendo o robô fixo com tiros e uma versão alternativa que girava no próprio eixo, um robô vigilante que não reagia ataques, exemplificado na Figura 5 a), e um robô agressivo que perseguia o jogador ao ser atacado ou aproximado, exemplificado na Figura 5 b). Após passar a porta, o gerador de níveis pode apresentar qualquer um desses inimigos, uma combinação deles ou nenhum deles. Alguns jogadores encontraram os robôs agressivos logo nos primeiros níveis, enquanto ainda estavam em fase de aprendizado dos controles e dos itens no cenário, morrendo várias vezes antes de conseguir algum avanço significativo. O jogo possuía também uma mecânica de níveis por habilidade, mas isso foi pouco explorado pelos jogadores. Apesar desse problema, os jogadores ainda gostaram da narrativa e a navegação diferente do jogo, talvez interessados em explorar um pouco mais se a dificuldade fosse melhor ajustada.

Com este retorno dos alunos, sabemos que não há um estilo preferido entre o puzzle ou a ação com os mecanismos de física, porém é necessário alguma coisa que chame a atenção a primeira vista, para estimular a experimentação. Uma interface clara e mecânica simples também ajuda o jogador a ter mais vontade de tentar e errar, mas deve-se evitar fazê-lo começar tudo do zero, para que não se torne frustrante. Por último, a dificuldade entre níveis deve ser condizente com o nível de habilidade do jogador, como já teorizado no Canal de Flow e confirmado pelo trecho do experimento utilizando o jogo Newton's Law.

JOGO DE FÍSICA RAMPA INCLINADA

Inicialmente, o conceito para o desenvolvimento do jogo era utilizar um circuito para levar um objeto de um lado a outro e peças que alterariam o movimento do objeto, como aumento de atrito e impulso no centro gravitacional. No entanto, essa proposta apresentava uma série de problemas de simulação, que deveria incluir apenas as três Leis de Newton, e, ao mesmo tempo, não parecer artificial. Após a observação, a ideia foi simplificada para ter como assunto principal os fenômenos de atrito, que é um fenômeno de fácil simulação com uma física personalizável pelos desenvolvedores. Essa simplificação facilitou para que o fenômeno fosse mais similar à realidade e, ao mesmo tempo, também simplificou as mecânicas para apenas uma variável influenciável para a experimentação.

Para abordar o conteúdo corretamente, foi necessário descobrirmos qual atividade de aprendizagem melhor difundia o conteúdo de atrito. Observou-se como sites com simulações educacionais apresentavam esse conteúdo, uma das mais fortes influências foram as simulações do PhET[11], e chegou-se a um padrão de um objeto de um material deslizando sobre outro material, que podem ser alterados para explorar os diferentes tipos de atrito entre materiais.

Combinando as duas ideias, criou-se um jogo cunhado como “Rampa Inclínada”, onde o cenário apresenta um objeto de material neutro e invariável, em cima de uma plataforma inclinável, cujo material varia de nível para nível (Veja exemplo da Figura 6, onde o material é concreto). e possui um coeficiente de atrito diferente informado no início de cada nível. O objetivo do jogo é utilizar os conhecimentos sobre o atrito para inclinar a rampa onde a caixa se encontra e levar o objeto em segurança até a plataforma, que possui diferentes faixas coloridas para uma pontuação diferenciada, estimulando o jogador a tentar atingir a faixa central, ao invés de contentar-se com as faixas das bordas.

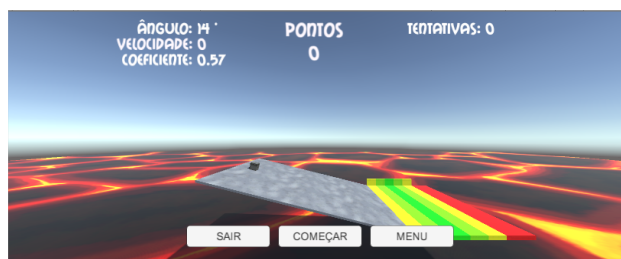


Figura 6. Exemplo de Nível do jogo Rampa Inclínada



Figura 7. Tela de sucesso e detalhe das faixas coloridas na plataforma destino.

Com o aprendizado do experimento de design centrado no usuário, a interface é simples e destaca os objetos utilizados para o jogo. Apesar de utilizar gráficos 3D, o fundo contrasta com os elementos que fazem parte das regras do jogo e evitam que o jogador perca o foco do que realmente é possível interagir na tela. Na rampa, 5 faixas que variam do verde (centro) até o amarelo (2 faixas externas), conforme pode ser visto na Figura 7, tornam fácil o entendimento de qual é a posição mais desejável para a caixa, com uma faixa vermelha adicional delimitando o limite antes que a caixa saia da rampa e também uma falha no nível.

A dificuldade entre diferentes níveis é apresentada como alteração do atrito, que trará um fenômeno diferente do observado no nível anterior e exigindo um pouco de experimentação e raciocínio pelo jogador. Com um coeficiente de atrito diferente, que será mostrado não só ao iniciar o nível como a aparência diferente da rampa que será constituída por outro material, a inclinação exigida pela rampa para levar o objeto até a plataforma de pontuação será diferente. O aluno pode apenas experimentar e ver os resultados de diferentes inclinações, ou pode usar uma equação para estimar qual inclinação irá levar a caixa até a aceleração necessária para chegar ao centro da rampa. Há pretensões de que haja uma premiação adicional para um bom resultado com poucas tentativas, mas isso ainda não foi implementado na versão apresentada neste trabalho.

Por fim, a mecânica pode não trazer algo diferente do conhecido pelo aluno de física, mas houve uma preocupação extra em tornar a simulação fiel ao que aconteceria na realidade sem utilizar-se de muitos

conceitos que o jogador ainda não conhece teoricamente. Para isso, o objeto utilizado é um objeto cúbico, que desliza sem torques pela rampa. Há um pequeno giro no final da rampa que se encontra com a plataforma de pontuação, mas ele não afeta a simulação de forma significativa para os fins de validação teórica. E o coeficiente de atrito auxilia o jogador a ligar o que está sendo apresentado na tela com o que lhe foi ensinado em sala de aula, ainda que ele não deseje utilizar as formas durante o jogo.

O jogo atual possui 4 níveis de dificuldade, com um planejamento para expandir para mais níveis com mais materiais e um caminho mais complexo para aumentar a dificuldade.

CONCLUSÃO

O envolvimento dos jogadores em uma fase inicial do design possibilitou o entendimento da visão que esses usuários tinham sobre jogos que utilizavam física como sua mecânica principal e, com isso, utilizar ou evitar características que foram citadas como boas ou ruins, respectivamente. Entretanto, o experimento não garante que o jogo desenvolvido irá agradar todos os usuários do mesmo tipo de público, apenas tornou o design mais próximo do ideal para este caso e trouxe uma visão interessante sobre o design centrado no usuário junto com design de jogos.

Para trabalhos futuros, esse jogo deverá ser reavaliado com jogadores mais próximos do público-alvo, alunos do 1º ano de Engenharia cursando a matéria de Física I, em outro experimento de observação, não apenas para validação do design, mas para refinar designs específicos que serão criados para um comportamento adaptativo com base em uma teoria de estilos de aprendizagem, incluindo mais uma iteração no trecho inserido no ciclo de design de games do Jesse Schell.

REFERENCES

1. ABRAS, C.; MALONEY-KRICHMAR, D.; PREECE, J. User-centered design. Bainbridge, W. Encyclopedia of Human-Computer Interaction. Thousand Oaks: Sage Publications, v. 37, n. 4, p. 445-456, 2004.
2. BAUER, M. W.; GASKELL, G. Pesquisa qualitativa com texto, imagem e som: um manual prático. In: Pesquisa qualitativa com texto, imagem e som: um manual prático. Vozes, 2010.
3. CRAWFORD, C. The Art of Computer Game Design. Osborne/McGraw-Hill Berkeley, CA, USA, 1984, 120 pag.
4. DRAWTOPIA, Super Smith Bros LTD. Disponível em <https://play.google.com/store/apps/details?id=com.supersmithbros.drawtopia&hl=pt_BR> Acesso em 10 de setembro de 2016
5. DUH, H. B. et al. A narrative-driven design approach for casual games with children. In Proceedings of the 5th ACM SIGGRAPH Symposium on Video Games (Sandbox '10), Stephen N. Spencer (Ed.). ACM, New York, NY, USA, 2010, 19-24.
6. KOSTER, R. Theory of Fun for Game Design. 2a Edição. O'Reilly Media, Inc. 2013, 256 pag.
7. MECANIKA, Games for Science. Disponível em <<http://www.gameforscience.com/physica/>> Acesso em 10 de setembro de 2016.
8. NEWTON'S LAW, Just Two. Disponível em <<http://www.crazygames.com/game/newtons-law>> Acesso em 10 de setembro de 2016
9. NORMAN, D. A. O design do dia-a-dia, Capítulo 7. Rocco, 2006.
10. PITUBA, J. L.. Adaptatividade em jogos (resumo). Em: Memórias do X Workshop de Tecnologia Adaptativa - WTA 2016. EPUSP, São Paulo. ISBN: 978-85-86686-86-3, p. 6. 28 e 29 de Janeiro, 2016.
11. PhET Interactive Simulations. Disponível em <<https://phet.colorado.edu/>> Acesso em 07 de Outubro de 2016.
12. PRENSKY, M. Aprendizagem Baseada em Jogos Digitais. São Paulo: SENAC, Ed. 1, 2012, 576 pag.
13. SCHELL, J. The art of game design: a book of lenses. Morgan Kaufmann Publishers Inc. San Francisco, CA, USA, 2008, 489 pag.
14. SICART, M. Defining game mechanics. Game Studies, v. 8, n. 2, p. 1-14, 2008. disponível em <<http://gamestudies.org/0802/articles/sicart>> Acesso em 10 de setembro de 2016.
15. ZIMMERMAN, E. e SALEN, K. Rules of Play: Game Design Fundamentals. The MIT Press, 2003, 688 pag..

Jogos Sérios Digitais para a Promoção do Património Natural

o caso de um jogo de exploração de um parque na busca dos seus animais

Liliana Santos¹, Daniel Pereira², Rui Nóbrega³, Pedro Beça⁴ e António Coelho⁵

¹e ⁴UA, Universidade de Aveiro

^{2,3}e ⁵FEUP, Faculdade de Engenharia da Universidade do Porto

³e ⁵INESC TEC, Instituto de Engenharia de Sistemas e Computadores – Tecnologia e Ciência

ABSTRACT

O presente trabalho visa, dentro de um contexto de divulgação científica num parque biológico, a criação de uma aplicação móvel enquanto ferramenta para disseminar conteúdos científicos. Apresenta-se um dos elementos dessa aplicação: o "detetor virtual de animais", as mecânicas do jogo, assim como a sua avaliação. Conclui-se que os utilizadores ficaram satisfeitos com o funcionamento do radar do jogo e reflete-se sobre possíveis melhorias.

INTRODUÇÃO

Nos dias de hoje, existem jogos que vão muito para além da sua componente lúdica, sendo também úteis para a promoção do turismo (Kachniewska, 2015), para a realização de simulações militares ou médicas (Michael & Chen, 2006), para o ensino (Jantke & Spundflasch, 2013) e ainda para a divulgação científica (Santos, 2016), entre outros. O jogo a ser apresentado neste artigo faz parte de um projeto de divulgação científica, no qual se pretende desenvolver uma aplicação para a divulgação, de forma interativa, do património natural com informação geolocalizada. A aplicação está a ser desenvolvida em parceria entre a Universidade de Aveiro, a Faculdade de Engenharia da Universidade do Porto e o Parque Biológico de Gaia, local piloto para testar a aplicação móvel. Esta inclui um mapa do parque com pontos de interesse, informação geolocalizada graças à integração de tecnologias de georreferenciação (GPS) e mini jogos temáticos.

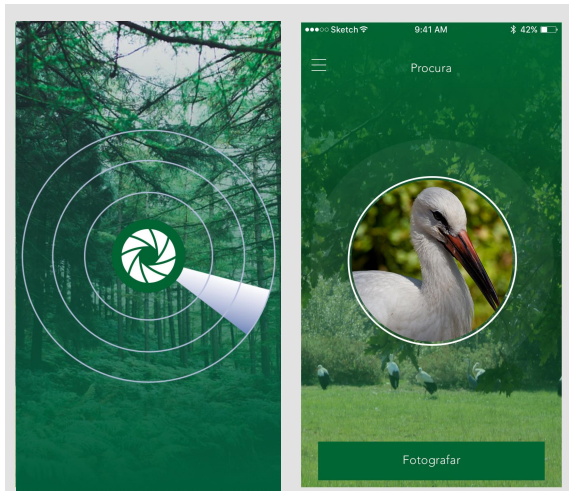


Figura 1. Mockups da interface gráfica do jogo: Radar, Detecção de animal.

Um segundo toque no ecrã faz aceder a mais informações sobre a espécie "encontrada". Um terceiro passo permite abrir uma galeria de imagens: a sua coleção de animais já registados. Ao clicar num animal da galeria, o jogador pode voltar a consultar as informações sobre esse animal, estando estas sempre acessíveis desde o momento em que o animal é registado (ver Figura 2).

Para avaliar e testar o jogo, vários voluntários experimentaram-no no Parque Biológico de Gaia. Depois do teste, cada participante preencheu um questionário. Uma das questões colocadas pedia que o utilizador indicasse o seu grau de satisfação com o funcionamento do radar.

Demo: <https://youtu.be/1dEO6qi9zs4>

RESULTADOS

58% dos participantes ficaram satisfeitos com o funcionamento do radar e 21% dos participantes, muito satisfeitos.
16% dos participantes afirmaram estarem pouco satisfeitos.

Foi mencionada a falta de uma legenda que explicasse o seu funcionamento. A falta de *feedback* do radar também foi observada ao longo da experiência por parte de outros participantes. Foi sugerido o uso de animações como forma de tornar o ecrã mais interativo. O tamanho da amostra foi reduzido, não permitindo tirar conclusões definitivas, mas pode, no entanto, fornecer algumas pistas para futuros melhoramentos.

This project was developed in the context of FourEyes, part of the project "TEC4Growth - Pervasive Intelligence: Enhancers and Proofs of Concept with Industrial Impact/NORTE-01-0145-FEDER-000020", financed by the North Portugal Regional Operational Programme (NORTE 2020), under the PORTUGAL 2020 Partnership Agreement, and through the European Regional Development Fund (ERDF).

Uma das funcionalidades da *app* é um "detetor virtual de animais". O objetivo é dar a conhecer as espécies animais que vivem em liberdade no parque, sensibilizando para o património natural.

O detetor virtual de animais

É um jogo de registo e colecionismo de animais existentes no Parque. A ideia é que o jogador, ao passear pelo parque, encontre animais virtuais que podem ser registados e colecionados, tendo acesso às fichas informativas dessas espécies.

Para tal, o jogador, quando passa nas proximidades de um par de coordenadas pré-definidas, recebe uma notificação no mapa do parque, inserido na aplicação. Para isso, existe um radar que deteta a aproximação do turista às coordenadas onde um animal costuma ser encontrado, enviando-lhe uma notificação (ícone intermitente e sinal sonoro). Caso essa espécie ainda não esteja registada na galeria, o utilizador tem a possibilidade de a registar ao tocar na imagem que aparece no ecrã, criando assim uma "fotografia" virtual do animal, que irá enriquecer a sua coleção (ver Figura 1.).

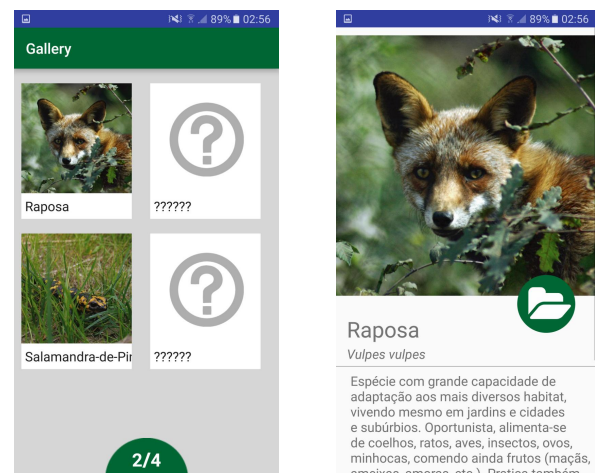


Figura 2. Mockups da interface gráfica do jogo: Galeria e Ficha Informativa.

CONCLUSÕES

Os testes de utilizador permitiram encontrar alternativas às atuais mecânicas de funcionamento.

O radar detetor de animais em liberdade presente na aplicação poderia funcionar como um radar verdadeiro, sendo que o utilizador poderia consultá-lo sempre que quisesse, a fim de descobrir se um animal do jogo se encontraria perto.

Também seria interessante que este radar fizesse uso do giroscópio e do acelerómetro para registar ou "fotografar" o animal encontrado. Esta é uma possibilidade a ponderar, visto que muitos *smartphones* já possuem essas ferramentas. Outra possibilidade é a atribuição de recompensas (descontos, *merchandising* ou material didático), em articulação com o Parque Biológico de Gaia, quando o jogador completasse a coleção.

Kachniewska, M. (2015). Gamification in tourism: pitfalls and benefits (in English) [w:] Tourism in the age of transformation, Economic University of Varna, Science&Economics, Varna 2015, pp. 399-407, ISBN 978-954-21-0864-1.

Jantke, K. P., & Spundflasch, S. (2013). Storyboarding Pervasive Learning Games. *Proceedings of the 2013 International Conference on Advanced Ict and Education*, 33(Icaicte), 42–48. <http://doi.org/10.2991/icaicte.2013.9>

Michael, D., & Chen, S. (2006). *Serious Games: Games That Educate, Train, and Inform*.

Santos, L. A. Da R.; Beça, P.; Correia, F. (2016). A Ilustração Científica Em Jogos Sérios Eletrónicos. V Encontro Brasileiro sobre Ilustração Científica.



INMENSION

TEM DESCUBRIR A REALIDADE AUMENTADA

Gregory Caldeira | José Dias | André Almeida | Ricardo Xavier
Ana Isabel Veloso | Hélder Caixinha | Pedro Amado
DeCA – Universidade de Aveiro

Em que consiste

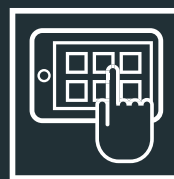
A INMENSION contém seis jogos interativos em realidade aumentada: Maze, Roll Ball, Stack, Random Balls, Pacman e Arkanoid. Para os descobrir, o jogador deverá formar uma equipa de pelo menos duas pessoas e depois procurar os identificadores (marco ou fiducial) para jogar.

Como funciona?



Descarrega a App

<http://inmension.web.ua.pt/html/>



Escolhe o teu jogo



Procura o identificador
(marco ou fiducial)

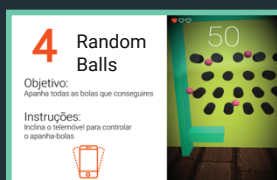
Abstract

A INMENSION é uma aplicação móvel que integra seis jogos interativos em realidade aumentada com objetivo de proporcionar uma divertida visita guiada aos novos públicos (escolas, novos alunos, conferências) que visitam o Departamento de Comunicação e Arte da Universidade de Aveiro.

Palavras-chave

- Realidade Aumentada
- Exploração Física
- Aplicação móvel
- Jogos interativos

Jogos



DeCA | ua | departamento de comunicação e arte
universidade de aveiro



universidade
de aveiro

FORGOTTEN SHAFT

Bruno Lima² Eduardo Galdino² Rennan Raffaele² Frutuoso Silva^{1,2}

1 Instituto de Telecomunicações

2 Universidade da Beira Interior

Resumo

Gillan, um explorador filho de oceanógrafos famosos, parte em uma aventura em busca dos seus pais que desapareceram num lugar chamado Forgotten Shaft e agora ele tenta encontrar alguma pista sobre o desaparecimento dos pais.

Como Funciona?

Forgotten Shaft é um jogo que combina elementos de **exploração** com a **realidade virtual** para proporcionar uma bela e imersiva experiência nesse universo marinho. Para jogar é necessária a utilização do Oculus Rift para observar o ambiente e um controle de PS4 para interação com o jogo. Explore esse oceano, tome cuidado com animais marinhos e encontre runas para guiar o seu caminho pelo templo de uma antiga civilização...

Um templo... runas... escuridão...
"relato de um viajante"



Agradecimentos: O autor filiado ao Instituto de Telecomunicações agradece o suporte dado através do programa FCT projeto UID/EEA/50008/2013.



Frutuoso Silva e Pedro Santos (eds.)
ISBN: 978-989-20-7148-0