

Universidade Federal do Pampa

Vinícius da Silva Lopes

Padrões para Interação e Usabilidade em Jogos MOBA e FPS

Alegrete

2017

Vinícius da Silva Lopes

Padrões para Interação e Usabilidade em Jogos MOBA e FPS

Trabalho de Conclusão de Curso apresentado ao Curso de Graduação em Engenharia de Software da Universidade Federal do Pampa como requisito parcial para a obtenção do título de Bacharel em Engenharia de Software.

Orientador: Jean Felipe Patikowski Cheiran

Alegrete

2017

Vinícius da Silva Lopes

Padrões para Interação e Usabilidade em Jogos MOBA e FPS

Trabalho de Conclusão de Curso apresentado ao Curso de Graduação em Engenharia de Software da Universidade Federal do Pampa como requisito parcial para a obtenção do título de Bacharel em Engenharia de Software.

Trabalho de Conclusão de Curso defendido e aprovado em de de

Banca examinadora:

Jean Felipe Patikowski Cheiran
Orientador

João Pablo Silva da Silva
UNIPAMPA

Marcelo Resende Thielo
UNIPAMPA

*Este trabalho é dedicado às pessoas que amo,
que sempre me apoiaram e continuam apoiando,
seja com um gesto de carinho e amor ou me colocando no caminho certo.*

*"Because of our weakness, we trained our eyes, ears and the ability to think.
Learning how to survive, that's our trait! We don't have any extraordinary senses.
But because we are weak, by learning and gaining experience,
we gained the wisdom to achieve the unachievable —to predict the future!—
We are humans; we are the most talented people!
Precisely because we were born without any ability, we can achieve anything."
(Yuu Kamiya, No Game No Life)*

Resumo

A utilização de padrões em jogos eletrônicos é uma das formas de garantir certo nível de aceitação no mercado. Empresas desenvolvedoras tendem a ter dificuldade de apresentar formas de interface e interação satisfatórias aos usuários que atendam todas as suas expectativas, podendo levar, por parte do usuário, ao abandono do jogo. O objetivo do presente trabalho é a criação de um catálogo de padrões para usabilidade e interação em jogos, no intuito de ajudar à criação de jogos melhores, evitando portanto cometer erros recorrentes onde já se tem estruturas bem definidas para correção. Para chegar a este catálogo, utilizam-se catálogos já existentes, avaliações heurísticas aplicadas dentro de jogos selecionados dentro de uma lista para identificar problemas e verificar sua recorrência em outros jogos de uma lista, além de analisar as soluções para esses mesmos problemas em jogos similares. O produto final inclui soluções descritivas para empresas desenvolvedoras que tenham dificuldade de apresentar essas formas de usabilidade e interação.

Palavras-chave: Usabilidade. *Design* de interface. Padrões de interação. Padrões em jogos

Abstract

The use of patterns in video games is a way of ensuring a certain level of market acceptance. Development companies tend to have difficulty delivering satisfactory forms of interaction and interfaces to users, to meet all their expectations, fact that can lead users to abandon the game. The goal of this study is to create a catalog of usability and interaction patterns in games, in order to help create better games, avoiding therefore, committing recurring errors for which already have well-defined structures for correction. To get to this catalog the author uses existant catalogs, heuristic evaluation to identify problems and verify its reoccurrence in a list of games, in addition to analyse possible solutions on similar games and their descriptions. This paper includes descriptive solutions for developing companies that have difficulty presenting forms of usability and interaction.

Key-words: Usability. Interface Design. Interaction Patterns. Game Patterns

Lista de ilustrações

Figura 1 – Gráfico modelo para jogos MOBA	27
Figura 2 – Tela inicial do site de pesquisa Google	31
Figura 3 – Exemplo de padrão de paginação no site Google	32
Figura 4 – Imagem retirada do jogo Defense of the Ancients: Allstars.	40
Figura 5 – Imagem retirada do jogo League of Legends.	40
Figura 6 – Imagem retirada do jogo Heroes of Newerth.	41
Figura 7 – Imagem retirada do jogo Defense of the Ancients 2 (DotA2).	41
Figura 8 – Imagem retirada do jogo Heroes of the Storm.	42
Figura 9 – Imagem retirada do jogo Magika: Wizard Wars.	43
Figura 10 – Imagem retirada do jogo Smite.	43
Figura 11 – Imagem retirada do jogo TOME: Immortal Arena.	44
Figura 12 – Imagem retirada do jogo Strife.	44
Figura 13 – Imagem retirada do jogo Arena of fate.	45
Figura 14 – Imagem retirada do jogo Smite Orcs Must Die! Unchained.	45
Figura 15 – Imagem retirada do jogo Guardians of Middle-Earth.	46
Figura 16 – Imagem retirada do jogo Supernova.	46
Figura 17 – Imagem retirada do jogo Doom.	47
Figura 18 – Imagem retirada do jogo Doom 2.	48
Figura 19 – Imagem retirada do jogo Doom 3.	48
Figura 20 – Imagem retirada do jogo Counter-Strike 1.6.	49
Figura 21 – Imagem retirada do jogo Battlefield 3.	49
Figura 22 – Imagem retirada do jogo Counter-Strike: Global Offensive.	50
Figura 23 – Imagem retirada do jogo Bioshock.	50
Figura 24 – Imagem retirada do jogo Bioshock Infinite.	51
Figura 25 – Imagem retirada do jogo Left 4 Dead.	51
Figura 26 – Imagem retirada do jogo Duke Nukem: 3D.	52
Figura 27 – Imagem retirada do jogo F.E.A.R.	52
Figura 28 – Imagem retirada do jogo Far Cry 3.	53
Figura 29 – Imagem referente a existência de vários indicadores de vida em múltiplos jogos.	57
Figura 30 – Imagem referente a existência de um minimapa em múltiplos jogos.	58
Figura 31 – Utilização do padrão “movimento Prioritário” explicitada por “can’t be disabled”	60
Figura 32 – Padrão Dificuldade de Nível Adaptativa	62
Figura 33 – Utilização de “Tutoriais para Iniciantes” em Defense of the Ancients 2.	63

Figura 34 – Utilização de “Visibilidade de Mapa Ampliada” em Guardian of Middle-Earth e Battlefield 3.	65
Figura 35 – Utilização de “Linguagem Limpa” em Left4Dead.	66
Figura 36 – Utilização de “Uso de HUD Personalizável” em League of Legends	67
Figura 37 – Utilização de “Teclas de Atalho Customizáveis” em Far Cry 3.	68
Figura 38 – Utilização de “Alerta de Erro/Bug” em League of Legends.	69
Figura 39 – Tela de Resumo em League of Legends para Verificação de Desempenho de jogador.	71
Figura 40 – Tela de Pós-jogo em League of Legends para Verificação de Desempenho de jogador.	72
Figura 41 – Tela de estatísticas em Pós-jogo de League of Legends para Verificação de Desempenho de jogador.	72
Figura 42 – Utilização de dupla barra de vida em DotA2.	74
Figura 43 – Utilização de “Sistema de Nivelamento de Partidas Online” em Counter-Strike: Global Offensive.	76
Figura 44 – Utilização de “Múltiplos Indicadores de Status Negativo de Personagem” em Defense of the Ancients 2.	77
Figura 45 – Utilização de “Reutilização de Configuração Prévia” em Heroes of the Newerth.	78
Figura 46 – Interface referente a fase <i>beta</i> do jogo League of Legends	80
Figura 47 – Interface referente aos primeiros anos do jogo League of Legends	80
Figura 48 – Interface atual do jogo League of legends	81

Lista de tabelas

Tabela 1 – Padrão Weenie Chain	55
Tabela 2 – Padrão Privileged Move	56

Lista de siglas

CAPES Coordenação de Aperfeiçoamento de Pessoal de Nível Superior

DotA Defense of the Ancients Allstars

DotA2 Defense of the Ancients 2

E3 Electronic Entertainment Expo

FPS First-Person Shooter

HUD Heads-up Display

IHC Interação Humano-Computador

MMORPG Massively Multiplayer Online Role-Playing Game

MOBA Massive Online Battle Arena

RPG Role-Playing Game

RTS Real Time Strategy

Sumário

1	INTRODUÇÃO E MOTIVAÇÃO	21
1.1	Objetivos	22
1.2	Organização do Trabalho	22
2	FUNDAMENTAÇÃO TEÓRICA	25
2.1	Jogos	25
2.1.1	Jogos digitais	25
2.1.2	Jogos MOBA	26
2.1.3	Jogos First-Person Shooter (FPS)	27
2.2	Interação Humano-Computador	28
2.2.1	Padrões	28
2.2.1.1	Padrões de usabilidade	29
2.2.1.2	Padrões de interação	31
3	TRABALHOS RELACIONADOS	33
3.1	The Case For Game Design Patterns	33
3.2	A Pattern Approach to Interaction Design	33
3.3	Easy to Use and Incredibly Difficult: On the Mythical Border between Interface and Gameplay	34
3.4	Heuristics and Usability Guidelines For The Creation And Evaluation Of Fun In Video Games	35
3.5	From Playability to a Hierarchical Game Usability Model	35
3.6	Usability Patterns in Games	36
3.7	Considerações do Capítulo	37
4	METODOLOGIA	39
4.1	Apresentação dos jogos escolhidos	39
4.1.1	Lista de jogos pelo gênero MOBA	39
4.1.2	Lista de jogos pelo gênero FPSs	47
4.2	Gabarito para catalogar padrões	53
4.3	Identificação de padrões já catalogados	54
4.4	Identificação de padrões ainda não catalogados	56
5	RESULTADOS	59
5.1	Padrões Refatorados	59
5.1.1	Priorização de Ações Específicas	59

5.1.2	Padrão Dificuldade de Nível Adaptativa	61
5.1.3	Padrão Tutoriais para Iniciantes	62
5.1.4	Padrão Visibilidade de Mapa Ampliada	64
5.1.5	Padrão Uso de Linguagem Limpa	65
5.1.6	Padrão Uso de HUD Personalizável	66
5.1.7	Padrão Teclas de Atalho Customizáveis	67
5.1.8	Padrão Uso de Alerta de Erro/Bug	68
5.2	Padrões Identificados	70
5.2.1	Padrão Verificação de Desempenho de Jogador	70
5.2.2	Padrão Múltiplos Indicadores de Vida	72
5.2.3	Padrão Sistema de Nivelamento de Partidas Online	74
5.2.4	Padrão Múltiplos Indicadores de Status Negativo de Personagem	76
5.2.5	Padrão Reutilização de Configuração Prévia	77
6	CONSIDERAÇÕES FINAIS	79
6.1	Descobertas e Contribuições	79
6.2	Limitações e Trabalhos Futuros	81
	REFERÊNCIAS	83
	Índice	85

1 Introdução e Motivação

A ideia de padrões de *design* se origina dos arquitetos [Alexander, Ishikawa e Silverstein \(1977\)](#) que, segundo eles, visava ampliar conhecimentos, gerando assim soluções esclarecíveis a serem usadas novamente dada sua taxa de sucesso. O mesmo conceito foi utilizado na computação, nas áreas de engenharia de software e Interação Humano-Computador (IHC). Na área de IHC, padrões são importantes por priorizarem a usabilidade e documentarem experiências de projetos anteriores para reuso. Em alguns casos, as interfaces de usuários apresentam arquitetura mal organizada, interações mal planejadas e inconsistências. Sendo assim, os padrões são vistos como formas de tornar boas soluções disponíveis para as pessoas que constroem as interfaces.

Por outro lado, existe a presença da usabilidade que, segundo [Nielsen \(1994\)](#), é um atributo que avalia o quão fáceis de usar são as interfaces, também se referindo a facilidade de utilização de protótipos de *design*. A necessidade da usabilidade não se limita apenas pelo *design*, sendo essa uma forma de sobrevivência no mercado atual como aponta [Folmer \(2006\)](#). Ainda falando de usabilidade, [Nielsen \(1994\)](#) ainda cita diversos fatores a serem analisados e alguns deles fortemente relacionado a jogos. Entre eles existe a capacidade de aprendizagem, que se entende por o quão intuitivo e fácil de lembrar será a manipulação dos elementos do jogo para chegar ao seu objetivo. O *gameplay*, ou jogabilidade, que é a maneira que um jogador interage com elementos dentro de um jogo, seja por meio de regras, ou conexão entre eles. Ele também influencia no *design* do jogo, sendo esse um ponto importante para o estudo de padrões de interação, já que a jogabilidade está ligada com o nível de diversão e subsequentemente a satisfação do usuário diferindo-se da usabilidade por ser restrita a visão do usuário.

Sendo assim, é visado identificar tais padrões em jogos [MOBA](#), do inglês *Massive Online Battle Arena* e [FPS](#), do inglês *First Person Shooter*, tal como o devido problema que gerou essa necessidade e a solução encontrada para saná-lo, apresentando também, exemplos de padrões encontrados e como eles são dispostos em diferentes jogos.

Tendo em vista que diferentes padrões são criados dependendo das necessidades dos jogadores. Jogos com diferentes temáticas ou estilos tendem a ter padrões diferentes, fazendo com que jogadores se adaptem a uma maneira de jogar. Ao deparar-se com novos tipos de interação, muitos usuários acabam parando de jogar por algum tempo ou até mesmo definitivamente, simplesmente por problemas de usabilidade do sistema. Os métodos adotados por empresas de jogos tendem a ser diferentes, dados os problemas encontrados e a necessidade de solução, tanto em seu estilo de interação como em seu *design*. Segundo [Folmer \(2006\)](#), padrões para jogos surgem a partir de aplicações anteri-

ores e principalmente limitações encontradas em jogos já produzidos. Esses paradigmas, ou modelos, surgem para solucionar problemas encontrados previamente, proporcionando assim, uma experiência mais profunda e empolgante para o jogador.

Essa necessidade de criar jogos melhores que, além de se mostrarem desafiadores, sejam consistentes e intuitivos, faz com que a observação e o conhecimento de técnicas previamente testadas, bem como a forma que o problema é resolvido com maneiras inovadoras, dêem suporte para a uma melhor aceitação deste produto no mercado atual. Esse conhecimento sendo passado de uma forma organizada torna a utilização dessas técnicas mais usual e de fácil acesso a empresas que queiram usufruir delas.

A importância de um estudo sobre a padronização de elementos de interface e interação, assim como padrões de usabilidade específicos para jogos se torna relevante, uma vez que padrões se tornam referência conforme sua taxa de sucesso. A utilização deles em jogos faz com que estudos referentes a essa área tenham um alto grau de importância para a satisfação do usuário final. Segundo estudo feito por [Federoff \(2002\)](#), empresas que se utilizam de técnicas que garantam uma melhor usabilidade e formas de interação, tendem a proporcionar maior diversão para o usuário.

Assim sendo, este trabalho utiliza heurísticas de usabilidade para identificar problemas de usabilidade e interação. Estes então são analisados, verificando-se uma possível solução para o determinado problema e finalmente catalogando essa solução em um formato padrão seguindo um gabarito apresentado no trabalho.

1.1 Objetivos

O objetivo desse trabalho é a catalogação de padrões de usabilidade e interação para jogos. Objetivos secundários incluem a formatação de padrões existentes conforme um gabarito e a identificação de padrões aplicados a jogos [MOBA](#) e [FPS](#).

Com base neste intuito, são utilizados jogos antigos e recentes, com sua devida explicação do porquê de estarem sendo utilizados. A partir deles, dá-se início a aplicação de avaliações heurísticas para a identificação de problemas de usabilidade que possam levar à identificação de padrões em jogos similares.

1.2 Organização do Trabalho

A divisão deste trabalho dá-se da seguinte forma: No segundo capítulo, a fundamentação teórica, visando a abstração do conteúdo em questão, tanto para a parte de jogos, para a parte de padrões das três áreas de conhecimento a serem estudadas.

No terceiro capítulo são apresentados trabalhos relacionados ao assunto em ques-

tão, onde é apresentada a ideia principal de cada autor e, subsequentemente, sua contribuição a esse trabalho.

No quarto capítulo é descrita a metodologia, apresentando o método para a construção desse trabalho, bem como a justificativa para a escolha dos jogos e, após apresentados, propostas as atividades futuras para este trabalho.

Seguido pelo quinto capítulo, onde são apresentados padrões já descritos e padrões propostos pelo autor. Além de, no sexto capítulo, serem apresentadas as considerações finais, contendo as limitações do trabalho e oportunidades de trabalhos futuros.

2 Fundamentação Teórica

Nesta seção é apresentada a explicação do conteúdo utilizado para a aplicação do trabalho, dividindo-se em duas grandes áreas, a área de jogos, justamente com a definição dos vários gêneros que a englobam, e a área de padrões de usabilidade e interação, onde é descrita sua funcionalidade e aplicação.

2.1 Jogos

Do latim "Jocus", ou diversão e brincadeira, jogos são toda e qualquer competição onde regras são criadas e bem definidas, constituindo um ambiente restrito. Segundo o filósofo [Huizinga \(2014\)](#), um jogo é:

“uma atividade voluntária exercida dentro de certos e determinados limites de tempo e espaço, segundo regras livremente consentidas, mas absolutamente obrigatórias, dotado de um fim em si mesmo, acompanhado de um sentimento de tensão e alegria e de uma consciência de ser diferente de vida cotidiana.”

Esses jogos normalmente envolvem estimulação física ou mental, tendendo a ser atividades estruturadas e praticadas com fins recreativos e pedagógicos, podendo ou não levar o jogador a vitória ou derrota.

Dentro do contexto deste trabalho, jogos devem ser interpretados como uma atividade regrada a fim de levar o jogador à conclusão deste, tendo em vista seus objetivos previamente especificados.

2.1.1 Jogos digitais

Jogos digitais, diferentemente de jogos comuns, estão ligados a aparelhos como computadores, celulares e videogames. Nesse sentido, são representações de jogos em um nível mais abstrato, através de recursos gerados computacionalmente. A possibilidade de criação de mundos alternativos os distingue de jogos físicos, e torna esses somente limitado a imaginação de seu criador e pela tecnologia usada. Outra característica marcante desse tipo de jogo é a rigidez de suas regras, pois em um espaço digital não se pode negociá-las.

Devido à possibilidade infinita de criação, jogos digitais são classificados a partir de algumas características, como: o objetivo do jogo, o contexto em que o jogador se encontra e a forma com que o jogador conduz seu personagem e interage com o ambiente. Segundo [Battaiola \(2000\)](#), a classificação desses jogos é:

- Estratégia: jogos cujas habilidades cognitivas são testadas; sua tomada de decisão decide sua capacidade de prosseguir e obter sucesso;
- Simuladores: jogos que tentam recriar ambientes, fazendo uma representação completa da física envolvida nesse; tendem a ser representações fieis e complexas do universo que os preenche;
- Aventura: jogos que utilizam enigmas implícitos, combinados com a necessidade de raciocínio lógico e interpretação de diálogos, normalmente apresentando uma história a ser seguida;
- Infantil: jogos destinados às crianças com a finalidade de divertir, e tendem a trazer assuntos educativos, mesmo que implícitos;
- Passatempo: jogos simples que desafiam o jogador através de quebra-cabeças de solução rápida que, em sua maioria, não possuem um enredo elaborado. Esses jogos são conhecidos também como jogos casuais e compreendem um gênero mais recente;
- Role-Playing Game (RPG): Jogo onde o usuário assume o papel de algum personagem e segue/cria narrativas para o mesmo. Adaptado de jogos de mesa, esse gênero foca na interação do usuário com o mundo imaginado, assim como em suas escolhas feitas neste mundo ao longo dessa narrativa;
- Esporte: jogos baseados em esportes reais, tais como futebol ou basquete;
- Educacionais: jogos que, possivelmente, se enquadram em outros grupos, mas sendo muito explícitos, utilizam critérios didáticos e pedagógicos sobre o assunto de que se trata o jogo.

Tendo em vista que esses gêneros são muito amplos, existem subgêneros, dos quais este trabalho trata, sendo esses: *Massive Online Battle Arena* ou **MOBA**, subgênero de jogos de estratégia e *First Person Shooter* ou **FPS**, gênero a par da lista de Battaiola (2000).

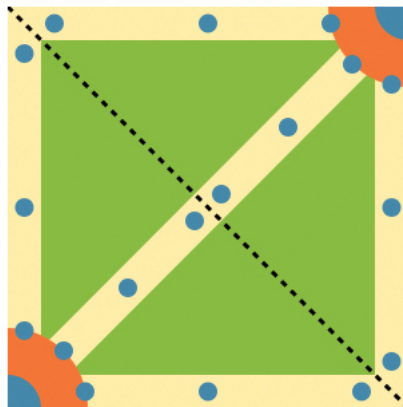
2.1.2 Jogos MOBA

Um jogo **MOBA** é um subgênero dos jogos Real Time Strategy (**RTS**), de estratégia em tempo real, que teve seu reconhecimento nos primeiros anos do século XXI com a aparição do jogo Defense of the Ancients Allstars (**DotA**) que se inspirava em um mapa modificado do jogo “Starcraft”.

Diferentemente de jogos **RTS**, o jogador, nos jogos **MOBA**, comanda uma única unidade (normalmente chamada de campeão ou herói) e a controla pelo cenário. Tendo como objetivo principal derrotar o time inimigo por meio da destruição do centro de sua

base (no caso do pioneiro [DotA](#)). Em um [MOBA](#), comumente é utilizado um *design* padrão para o mapa global baseado na figura 1. Esse contendo bases (laranja), que são utilizadas pelo jogador para fortalecê-lo e que devem ser protegidas de ataques do inimigo e vice-versa, além de comumente criarem tropas pré-programadas que ajudam o jogador a alcançar o objetivo principal; lanes ou linhas (amarelo), onde o jogador deve trilhar até chegar à base de seu inimigo, além de serem utilizadas pelas tropas como rota; torres (pontos azuis), que protegem a base ao longo da trilha e a *jungle* ou floresta (verde), onde há diferentes tipos de monstros a serem mortos pelo jogador para torná-lo mais forte, como demonstra a imagem a seguir (Figura 1). Esse *design* pode ser diferenciado, mas muitos desses elementos estão presentes em ambos.

Figura 1 – Gráfico modelo para jogos [MOBA](#)



Fonte: https://livrelinux.files.wordpress.com/2013/10/map_of_moba.png?w=676

Com a popularização dos [MOBAs](#) em 2010, o jogo “League of Legends” e hoje se tornou o jogo mais jogado no mundo, segundo pesquisa feita por [Gaudiosi \(2012\)](#), e permitindo que esse subgênero não pare de crescer.

2.1.3 Jogos [FPS](#)

Jogo [FPS](#) é um gênero de jogos focado em primeira pessoa e centrado em combate com armas. Diferente dos jogos [MOBA](#), esse gênero começou em meados dos anos 80, tendo sua aparição para o público em *arcades* segundo aponta pesquisa feita por [Bartholl \(2007\)](#), sendo essas, máquinas grandes que portavam somente um jogo. Ao longo do tempo esse gênero foi sendo adaptado para computadores o que expandiu ainda mais sua popularidade.

Jogos como Quake e Unreal, antigos e similares aos jogos *multiplayers* que temos atualmente, foram essenciais para o legado desse gênero, dado seu *design* e forma de

interação, assim como seus sucessores, como Half-Life, um jogo desenvolvido pela Valve, que dependia tanto do combate com armas como o raciocínio lógico para ultrapassar obstáculos e difíceis quebra-cabeças.

Com a evolução desse gênero, alguns jogos foram deixando de lado a mecânica do combate com armas, mas ainda sendo considerados FPSs. Um desses casos ocorre no jogo Mirror's Edge, onde o personagem passa a maior parte do jogo realizando acrobacias e saltos, dando a ilusão de movimento e emoção (frio na barriga). Quando seu combate com armas fica evidente, o jogo se utiliza de métodos já utilizados em muitos outros jogos, o que mantém sua característica como um jogo do gênero.

Desde então, a indústria desse gênero de jogos tem continuado a crescer, chegando à atualidade como um dos gêneros mais conhecidos e jogados pelo público, segundo aponta estudo feito por Bartholl (2007), tornando-o viável para o estudo da evolução de padrões de interação, bem como uma comparação relevante para a demonstração de diferentes padrões em diferentes gêneros de jogos.

2.2 Interação Humano-Computador

A IHC trata do estudo da comunicação que ocorre entre o ser humano e a máquina, essa podendo ser um computador, um celular ou qualquer outro dispositivo. Esse estudo está relacionado não somente com a ciência da computação e a engenharia do *software* mas com a psicologia, sociologia, semiótica, linguística, e áreas afins.

Essa comunicação se dá através de um *software*, como por exemplo uma interface na tela do computador ou um *hardware*, como por exemplo a utilização de um *mouse* para manipular e interagir com um dispositivo.

Com o constante crescimento da área tecnológica, formas de interação entre uma pessoa e uma máquina vão se tornando mais simples, seguras e agradáveis. Visando criar a viabilidade de interação com todos, não importando se a pessoa tenha algum tipo de deficiência. A capacidade humana de utilizar computadores é atualmente uma área de pesquisa e desenvolvimento muito abrangente e que se expande constantemente, já que esse estudo não utiliza somente ferramentas computacionais e *design* gráfico mas também de dados provenientes da psicologia experimental e educacional, além de contar com outros fatores humanos dentro da sociologia.

2.2.1 Padrões

O conceito de padrões na engenharia de software remete a estudos feitos por Gamma et al. (1994). Para eles esses padrões são soluções gerais reutilizáveis para problemas que ocorrem com frequência dentro do contexto de projeto de software. Posteri-

ormente, esse conceito foi expandido para a área de *design* de interação e de usabilidade, onde se incorporou a área de IHC.

2.2.1.1 Padrões de usabilidade

A usabilidade, segundo [Barbosa e Silva \(2010\)](#), se trata do quão intuitivo e de fácil manejo um sistema pode ser, remetendo tanto ao *design* de interface como o de interação. Na engenharia de software, ainda seguindo estudos feitos por [Barbosa e Silva \(2010\)](#), a usabilidade está associada tanto à qualidade quanto à IHC e visa garantir a eficiência e eficácia de um sistema. A eficiência, por sua vez, refere-se à interação produtiva entre o usuário e o sistema, permitindo a realização de tarefas com menor esforço. Já a eficácia pode ser entendida como a capacidade que um sistema tem de melhor apresentar sua interação, possibilitando ao usuário uma melhor abstração do que lhe é apresentado, assim finalizando a tarefa desejada.

Padrões de usabilidade remetem a problemas de usabilidade encontrados frequentemente em sistemas. Respostas a esses obstáculos são criadas e conforme sua taxa de sucesso vão aos poucos se tornando soluções gerais reutilizáveis. A garantia de cessão do problema faz com que este possa ser aplicado novamente em um contexto similar, podendo evoluir para um padrão.

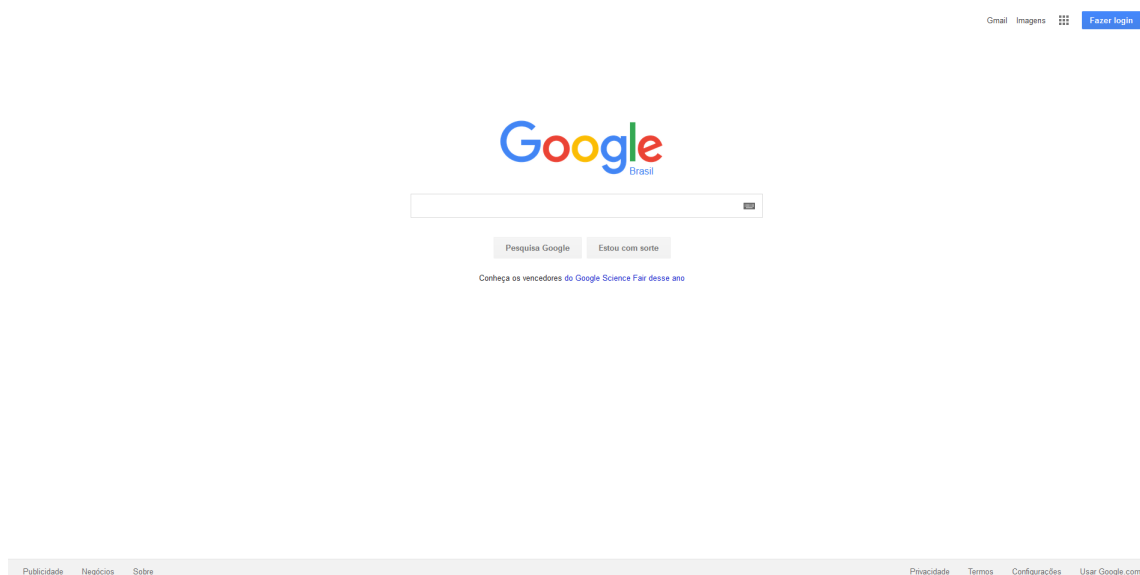
Segundo [Nielsen \(1994\)](#), devem ser utilizados alguns parâmetros para a avaliação de usabilidade para evitar erros que podem prejudicar a experiência de um usuário com um sistema. Esse tipo de avaliação segue as seguintes heurísticas:

- **Visibilidade de Status do Sistema:** Trata-se de verificar que a interface sempre mantém o usuário informado sobre o que está acontecendo por meio de *feedback* claro e fornecido em tempo razoável.
- **Relacionamento entre a interface do sistema e o mundo real:** A comunicação do sistema com o usuário deve conter uma ordem lógica e natural e evitar o uso de linguagem na qual o usuário não está habituado.
- **Liberdade de controle fácil pro usuário:** Facilitar as “saídas de emergência” para o usuário, permitindo desfazer ou refazer a ação no sistema e retornar ao ponto anterior, quando estiver perdido ou em situações inesperadas.
- **Consistência:** Utilizar da mesma linguagem e padrão de interação em todo o sistema para facilitar a assimilação do usuário.
- **Prevenção de erros:** Na tradução livre das palavras de [Nielsen \(1994\)](#) “Ainda melhor que uma boa mensagem de erro é um *design* cuidadoso que possa prevenir esses erros”. Ações definitivas, como deleções ou solicitações podem vir acompanhadas de um *checkbox* ou uma mensagem de confirmação.

- Reconhecimento ao invés de lembrança: Evitar sobrecarregar a memória do usuário o tempo inteiro, fazendo com que cada ação precise ser revista mentalmente antes de ser executada. Permitir que a interface ofereça ajuda contextual e informações capazes de orientar as ações do usuário – ou seja – que o sistema dialogue com o usuário.
- Flexibilidade e eficiência de uso: O sistema precisa ser fácil para usuários leigos, mas flexível o bastante para tornar-se ágil a usuários avançados. Essa flexibilidade pode ser obtida com a permissão de teclas de atalhos, por exemplo. No caso de *websites*, uso de máscaras e navegação com *tab* em formulários são outros exemplos.
- Estética e *design* minimalista: Os “diálogos” do sistema precisam ser simples, diretos e naturais, presentes nos momentos em que são necessários.
- Ajude os usuários a reconhecer, diagnosticar e sanar os erros: As mensagens de erro do sistema devem possuir uma redação simples e clara que ao invés de intimidar o usuário com o erro, indique uma saída construtiva ou possível solução.
- Ajuda e documentação: Evitar ao máximo a necessidade de ajuda na utilização do sistema. Ainda assim, um bom conjunto de documentação e ajuda deve ser utilizado para orientar o usuário em caso de dúvida. Deve ser visível, facilmente acessada, e com oferecer uma ferramenta de busca na ajuda.

Um simples exemplo de padrão de usabilidade que remete a uma das heurísticas de [Nielsen \(1994\)](#) seria a facilidade de interpretação de elementos básicos para usuários leigos. Informações apresentadas de forma clara e visível se tornam eficientes e levam o usuário a concluir a tarefa desejada em períodos menores de tempo, segue o exemplo na figura 2 abaixo destacada por [Toxboe \(Acesso em 14 de novembro de 2015\)](#) no site de pesquisa do Google, onde esse mecanismo de pesquisa fica evidente e, ao mesmo tempo consegue mostrar suas outras funcionalidades.

Figura 2 – Tela inicial do site de pesquisa Google



Fonte: O próprio autor

2.2.1.2 Padrões de interação

A interação se dá pela forma com que o usuário interage com um sistema, assim como a forma com que o sistema responde a essa interação. A melhoria da relação humano-computador proporciona uma melhor abstração por parte do usuário, gerando assim uma maior probabilidade de sucesso de um produto no mercado. Sistemas de fácil manejo tendem a ser mais receptivos, gerando benefícios para quem está utilizando-o e tendo como objetivo principal sempre proporcionar a máxima interatividade do produto da melhor forma possível.

A necessidade de tornar a interpretação de um usuário ao ver elementos de um máquina mais fácil e eficiente se volta para o conceito de interfaces, que também remete a interação com o sistema, mais precisamente a interação com o *design* centrado no usuário, onde o processo de *design* deve equilibrar funcionalidades e elementos visuais não apelativos. Essa área está presente em praticamente todos sistemas que possuem elementos visuais, desde computadores e celulares a carros e aviões. Um bom *design* de interface facilita a abstração e compreensão desses elementos e, até mesmo, minimiza o tempo necessário para o usuário concluir uma tarefa manualmente.

Padrões de interações avançam explorando paradigmas e tirando proveito dos avanços tecnológicos, enquanto a capacidade e complexidade dos dispositivos evoluem. A utilização de padrões garante qualidade e robustez, bem como a fácil utilização do sistema em todas as suas funcionalidades.

Ao aplicar soluções para determinados problemas de interação, criam-se produtos

e serviços de maior usabilidade, levando em conta os objetivos, funções, experiências, necessidades e desejos dos usuários. Essas soluções são comumente encontradas e podem evoluir em padrões, já que problemas desse tipo são documentados. A apresentação dessas soluções, seja para ampliar a usabilidade ou padronizar elementos de interface, confere ao produto uma maior compreensão de elementos visuais ou uma maior facilidade de uso.

Um simples exemplo de padrão de interação é a utilização de paginações, ou numeração de vários resultados possíveis dada a interação do usuário, como mostrado na figura 3. Para ordenar esses resultados, utiliza-se este padrão, já que a apresentação deles em uma única página seria inviável.

Figura 3 – Exemplo de padrão de paginação no site Google



Fonte: O autor

3 Trabalhos Relacionados

Este capítulo apresenta trabalhos e os relaciona a este. Devem ser apresentados trabalhos tanto sobre usabilidade quanto de interação com usuário, tendo em vista o objetivo de cada autor e a respectiva contribuição para com este trabalho.

3.1 The Case For Game Design Patterns

Em seu artigo, [Kreimeier \(2002\)](#) cita que, ao se utilizar de padrões, torna-se mais fácil prover uma solução para o problema descrito, fazendo com que não somente os usuários, mas também a própria equipe de design possa entender seus pontos de vista através do padrão. Para facilitar a criação e navegação por um padrão, Kreimeier adota um formalismo para o design de jogos, onde técnicas para diminuir a dificuldade de operação do sistema são criadas.

[Kreimeier \(2002\)](#) ainda cita a importância da diferenciação entre métodos baseados em padrões, que, em geral, são muito genéricos, dados padrões não específicos para jogos, e padrões específicos, que são coleções criadas com um propósito único, visando à solução de um problema específico. Para ele, utilizar-se de um padrão, não garantirá a inexistência de problemas, dado que padrões são, nada mais que receitas para se chegar a um final e não uma forma concreta de solução.

Em seu trabalho, Kreimeier cria um modelo para padronização de elementos de interação em jogos, além de mostrar alguns exemplos de padrões de interação. Entre eles está o padrão "*Weenie Chain*", que lida com o senso de direção do jogador, fazendo com que a falta de indicadores possam causar problemas para o jogador chegar ao seu objetivo final. Outro padrão apresentado por ele dentro do escopo deste trabalho é o padrão "*Privileged Move*", onde deve-se existir uma hierarquia para priorizar cada movimento ou interação, já que uma pode interferir na outra.

Este trabalho, por sua vez, cria um modelo para padronizar não somente elementos de interação, mas também a usabilidade de um jogo. Também apresenta uma comparação em diferentes gêneros de jogos e finalmente um catálogo com padrões de usabilidade e interação com o usuário.

3.2 A Pattern Approach to Interaction Design

Em se tratando de *design* e interação com interfaces, a dificuldade de abstração é um grande problema que sempre está presente, citado por [Borchers \(2001\)](#) em seu

artigo, como a necessidade de preencher a grande expectativa em relação à usabilidade. A dificuldade de chegar-se a uma interface que consiga expressar suas diretrizes, que são nada menos que instruções, é muitas vezes um dos principais problemas na área de IHC.

Para Borchers, a comunidade presente na área de IHC deve ser vasta e variada, aumentando o alcance e a importância dada a padrões de design interativo. Quando Borchers (2001) diz que: “Padrões têm sido uma forma bem sucedida na engenharia de software”, ele faz da utilização de padrões uma forma suscetível a ser aceita pelo público, visando o conhecimento do usuário final, a análise de produtos já existentes, a prototipação e o design interativo. Ressaltando ainda, a reutilização desses paradigmas pelo seu suscetível sucesso.

Para facilitar a criação e navegação por um padrão, Borchers também explica que deve haver uma forma sintática definida, criando-se um modelo para a padronização de elementos de interface.

A diferença do trabalho apresentado para este, dá-se pelo fato de que este trabalho, além de apresentar uma forma para padronização, apresenta uma comparação da disponibilidade de padrões em diferentes gêneros de jogos, juntamente com a catalogação de padrões não somente de usabilidade, mas também de interação com o usuário.

3.3 Easy to Use and Incredibly Difficult: On the Mythical Border between Interface and Gameplay

Segundo análise feita por Juul e Norton (2009), uma interface é a forma de interação do usuário com o jogo, mas não parte de sua jogabilidade. Para eles a interface deve ser de fácil utilização e a jogabilidade deve ser desafiante. Um jogo onde se ganha tudo muito fácil é logo esquecido e jogos que, mesmo conhecendo todas as informações expressas na tela, são difíceis de completar continuam a crescer aos olhos dos jogadores. Eles também citam uma frase usada por Malone (1982) que diz: “Jogos são fáceis de jogar, mas difíceis de serem completados com maestria”.

Os limites alcançados pela interface dificultam a imersão do jogador e, portanto, limitam a eficiência do jogo. A importância de utilização de interfaces interessantes e agradáveis para ajudar a satisfazer as necessidades do usuário. Em seu estudo, ele ainda analisa o interesse dos jogadores, dadas as várias formas de interação com o jogo, e algumas que se destacam são: o *feedback* dado ao usuário, a representação gráfica na tela e a música. Sistemas computacionais devem ser sensíveis aos usuários, sendo mais prazeroso ao uso e mais interessantes e finalmente, mais satisfatórios.

Juul e Norton (2009) tem como objetivo explicar e contextualizar a diferença entre problemas de usabilidade e a necessidade do jogo de aplicar certa dificuldade em

sua conclusão. Este trabalho difere-se do apresentado pelo fato de abranger não somente a usabilidade, mas também padrões de interação com usuário, além de apresentar diversos padrões organizados dentro de um catálogo e suas respectivas diferenças em diferentes gêneros de jogos.

3.4 Heuristics and Usability Guidelines For The Creation And Evaluation Of Fun In Video Games

Federoff (2002) em sua tese, considera os jogos como diferentes, logo não podendo se utilizar da definição global da palavra usabilidade. Ela cita também que um software comum é comprado e utilizado para fazer tarefas necessárias, mas que jogos são comprados com base em seu valor de entretenimento.

Se um jogo é divertido, logo ele causará a satisfação do usuário, encorajando assim o uso da usabilidade. Baseando-se em Nielsen (1994), ela atesta que jogos eficientes são aqueles que alcançam seu principal objetivo, sendo esse, o entretenimento, e que, para proporcionar esse objetivo, existe uma série de medidas a serem tomadas. Entre elas, o que ela chama de *funability*, ou a taxa de diversão observada nos usuários. Junto a esse conceito, Federoff sugere a avaliação de várias diretrizes, criando assim, uma tabela com varias heurísticas que no passado eram vistas como problemas de usabilidade, e hoje, padronizadas assim como Kreimeier (2002) cita, uma receita detalhada sobre uma solução de algum problema.

Após sua pesquisa, Federoff explica que, o termo usabilidade, em um jogo, lembra uma ferramenta para ajudar na produtividade e objetivo do mesmo. Interfaces e formas de interação são tópicos importantes quando o assunto é atingir a satisfação do usuário.

Diferentemente de Federoff, que almejava avaliar a quantidade de diversão envolvida em um jogo por meio de heurísticas de usabilidade, a apresentação do catálogo complementa ideias estudadas por Federoff no desenvolvimento de seu trabalho, além deste trabalho apresentar o termo usabilidade juntamente com outra área de conhecimento, a interação com usuário.

3.5 From Playability to a Hierarchical Game Usability Model

O termo *playability*, citado por Nacke (2009), define a jogabilidade na visão do jogador, o que estabelece uma ligação com a usabilidade. Para Nacke, a usabilidade se define por o uso de um método para se chegar com eficiência a um objetivo efetivo e satisfatório.

Assumindo que a jogabilidade está relacionada à medida de experiência, esta então,

dá suporte à usabilidade. O conhecimento de regras e funcionalidades gera a experiência do jogador, fazendo com que jogos similares tenham padrões de usabilidade similares, produzindo indiretamente padrões.

Esses aspectos tem uma relação direta com a forma de interação e o *feedback*, que é a resposta recebida pelo usuário ou a máquina após uma ação, fazendo com que jogadores se sintam mais à vontade em relação a jogos de mesmo estilo. Como seu objetivo, Nacke almejava criar e se utilizar de um modelo baseado na jogabilidade para fundamentar a usabilidade. Esse modelo mede desde o abstrato até o concreto, medindo também entidades relacionadas e propondo teorias e práticas para a usabilidade de uma aplicação.

O termo citado por Nacke tem forte ligação com um dos objetivos do trabalho, mas em contra partida, esse objetivo desse ser alcançado mediante outros, o conjunto de ideias apresentadas por esse trabalho complementa a ideia do autor em tornar um sistema efetivo e satisfatório.

3.6 Usability Patterns in Games

Ainda se tratando de padrões de usabilidade, [Folmer \(2006\)](#) implica que a indústria de jogos além de cara é de alto risco, pois a chance de sucesso de um jogo é muito baixa e que, para alcançar o sucesso, um jogo deve ter, entre outros fatores, a inovação na jogabilidade.

Ele também expõe que, a usabilidade é importante em jogos, assim como as formas de interação, pois se um jogador não compreender como se deve jogar, há um problema de usabilidade. Ele fala também que o design deve ser baseado em conhecimento já adquirido e ter seu sucesso comprovado, a arte do design de jogos tenta minimizar os riscos de problemas de usabilidade.

Folmer se utiliza de um modelo com sete elementos essenciais, sendo esses: o problema em questão; o contexto em que o problema se encontra; a força de impacto do problema; a solução; o princípio, que remete as heurísticas de usabilidade de Nielsen, assim sendo reconhecido e provado; o porquê de aplicar uma solução baseada em padrões de usabilidade; e por ultimo, exemplos, que são nada menos que aplicações anteriores que tiveram problema similar e se utilizaram de certo padrão para sanar o problema. Ao reconhecer um padrão, deve-se prová-lo, dados exemplos de utilização anterior.

A dificuldade encontra-se ao se deparar a um novo problema sem precedentes, o que leva a recorrer a padrões de usabilidade gerais, não especificados para jogos, tendo assim dificuldade de representação do problema e em sua gama de supostas novas soluções.

Juntamente com a usabilidade, outras áreas tem forte ligação com a taxa de sucesso de um jogo no mercado atual, este trabalho busca se utilizar de conhecimentos adquiridos

a partir das ideias de Folmer para, criar e catalogar padrões de usabilidade e interação, garantindo assim, uma maior probabilidade de sucesso, tanto para empresas quanto para a satisfação do usuário final.

3.7 Considerações do Capítulo

Relacionando a ideia de padrão de Kreimeier e Borchers, vê-se a diferença entre padrões de interação específicos para jogos e padrões de cunho geral, voltados para a engenharia de software. Padrões voltados a jogos devem ser utilizados a par do usuário final, pois isso determina o quão provável será a satisfação do usuário ao jogá-lo.

A necessidade de criar um ambiente imersivo a um jogador assim como a de tornar a interação com o jogo o intuitivo possível volta-se a utilização de táticas já conhecidas pelos usuários, citado por Folmer como a criação de novas soluções dados problemas anteriores. Essas táticas são comumente vistas em franquias de jogos, onde mantêm-se o que o usuário já esta acostumado, corrigindo pequenos detalhes que geraram problemas e adicionando novos elementos de interface, proporcionando assim uma experiência revigorante e ao mesmo tempo dando a sensação de inovação que o jogador tanto espera em novos jogos, segundo termo apresentado por Nacke.

Alcançar esse nível de satisfação é difícil já atestado por Federoff, mas se utilizando de técnicas de design já comprovadas e juntando inovação com formas já comprovadas de interação, dá-se mais possibilidade de sucesso para um futuro jogo.

4 Metodologia

Este capítulo apresenta a lista de jogos selecionados para ambos os jogos MOBA e FPS. Além de apresentar os critérios nos quais foram selecionados. Também apresenta o gabarito utilizado para a estruturação dos padrões, os métodos utilizados para a sua identificação tanto para aqueles já apresentados em outros catálogos, como para aqueles identificados pelo autor através de avaliações heurísticas.

4.1 Apresentação dos jogos escolhidos

A apresentação dos jogos se dá em dois grupos, dividindo-os pelo seus respectivos gêneros, [MOBA](#) e [FPS](#). Cada grupo contém sete jogos, cada um destacando-se por alguma mecânica inovadora que atendeu ou superou as expectativas dos jogadores, ou até mesmo jogos que foram utilizados como modelo de usabilidade, interação ou *design* gráfico, bem como jogos que ficaram na história do gênero, sendo usados para o estudo de problemas envolvendo essas três áreas de conhecimento. A escolha dos gêneros se deu pela sua popularidade atualmente, segundo Entertainment Software [Association \(2017\)](#). Além de serem gêneros com épocas de criação distintas: MOBAs a partir da metade dos anos 2000, e FPSs no começo dos anos 90. Já a escolha dos jogos se deu devido ao seu acesso gratuito ou já obtido anteriormente pelo autor, já que nenhum jogo foi comprado exclusivamente para a realização desse trabalho.

4.1.1 Lista de jogos pelo gênero [MOBA](#)

Defense of the Ancients: Allstars ou [DotA](#), adaptado de um mapa do Jogo de estratégia "Starcraft" e posteriormente do jogo "Warcraft III: Reign of Chaos", foi idealizado por um sócio da Empresa Blizzard no ano de 2005, sendo produzido de forma amadora. Esse jogo popularizou o gênero, com mecânicas que induziram muitos [MOBAs](#) a ser o que são hoje. Este jogo serve como um grande parâmetro no quesito evolução das áreas de conhecimento a serem estudadas. Segue-se uma imagem referente ao jogo (Figura 4).

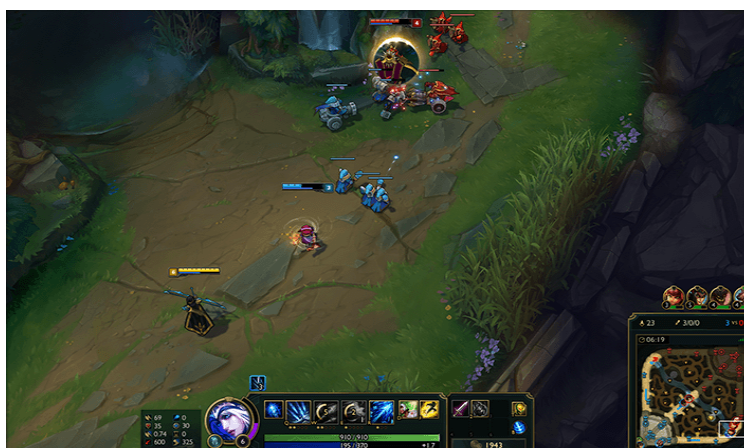
Figura 4 – Imagem retirada do jogo Defense of the Ancients: Allstars.



Fonte: O autor.

League of Legends: Desenvolvido e publicado pela Riot Games e lançado em outubro de 2009, o jogo passou por diversas atualizações para corrigir erros, mas também para atualizar tanto sua parte gráfica quanto a de interação. Diversas mudanças ocorreram durante esses poucos anos que o jogo está no mercado, muitas delas visando a usabilidade e a facilidade de interação e abstração de objetos na tela, como por exemplo as múltiplas atualizações de interface. Tomando como ponto de partida a versão beta do jogo até a atual, pode-se ter um ideia da evolução proporcionada nesses anos, bem como o surgimento de novas formas de facilitar a jogabilidade deste. Segue-se uma imagem referente ao jogo (Figura 5).

Figura 5 – Imagem retirada do jogo League of Legends.



Fonte: O autor.

Heroes of the Newerth: Desenvolvido pela S2 Games, o jogo foi lançado em maio de 2010, este foi baseado um mapa customizado para o jogo "Warcraft III: The Frozen Throne". Tem muitas semelhanças com o jogo [DotA](#). O que os diferencia são as atuali-

zações em jogabilidade, estatísticas e comunicação dentro do jogo. Utilizando-se de um motor gráfico feito pela própria empresa, esse jogo junta traços antigos desenvolvidos para outro jogo com ideias novas e formas inovadoras de *design* e interação, como por exemplo seus múltiplos estilos de jogo. Uma imagem do jogo pode ser observada na Figura 6.

Figura 6 – Imagem retirada do jogo Heroes of Newerth.



Fonte: O autor.

Defense of the Ancients 2 ou [DotA2](#): Desenvolvido pela Valve Corporation, dando seqüência ao [DotAs](#), este foi produzido por uma equipe desenvolvedora juntamente com a equipe original. Oficialmente, o jogo foi lançado em julho de 2013 e desde então tem ganhado espaço tanto no cenário casual quanto no competitivo, levando a prêmios de um milhão de dólares. Como seqüência de uma franquia, muitas ideias foram mantidas e atualizadas, dadas novas tecnologias emergentes, fazendo deste, um ótimo jogo para a possível descoberta de padrões. Segue-se uma imagem referente ao jogo (Figura 7).

Figura 7 – Imagem retirada do jogo [DotA2](#).



Fonte: O autor.

Heroes of the Storm: Desenvolvido pela Blizzard e lançado oficialmente em junho de 2015, o jogo vem passando por muitas melhorias de usabilidade, desde seu lançamento limitado em março de 2014, ainda em fase alfa para testes. Apresentando mecânicas inovadoras, assim como formas novas de interação, sendo uma delas a inexistência de itens de compra, focando em talentos do personagem. Esse jogo apresenta um grande potencial para o estudo de padrões ainda não catalogados. Uma imagem do jogo pode ser observada na Figura 8.

Figura 8 – Imagem retirada do jogo Heroes of the Storm.



Fonte: O autor.

Magika: Wizard Wars: Desenvolvido pela Paradox e lançado em maio de 2014, esse jogo possui além de mecânicas diferenciadas de qualquer outro **MOBAs**, elementos de jogos **RPGs top-down**, um exemplo desse tipo de **RPG** seria o famoso Diablo, jogo desenvolvido pela Blizzard. A utilização de elementos de **RPGs** com técnicas de jogos **RTSs**, faz deste jogo um bom ponto de partida para o surgimento de novos padrões, pelo fato de conseguir unir esses dois gêneros, bem como a diferenciação de disponibilidade de vários padrões em diferentes gêneros de jogos. Uma imagem do jogo pode ser observada na Figura 9.

Figura 9 – Imagem retirada do jogo Magika: Wizard Wars.



Fonte: O autor.

Smite: Desenvolvido pela Hi-Rez Studios e lançado em março de 2014, este jogo focado na personificação de deuses, se diferencia dos de mais pelo modo como é apresentado ao jogador, se utilizando de mecânicas muito vistas em MMORPGs. Ele assume a câmera em terceira pessoa tendo o *mouse* como "mira", diferente dos MOBAs normais, usualmente adotando o estilo *top-down*, esse jogo entra nesta lista como uma nova forma representativa dos jogos MOBAs. Segue-se uma imagem referente ao jogo (Figura 10).

Figura 10 – Imagem retirada do jogo Smite.



Fonte: O autor.

TOME: Immortal Arena: Desenvolvido pela KIXEYE e lançado em junho de 2014, é um jogo com menos objetivos secundários e focado no combate. Contendo gráficos atuais, o jogo entra para a lista por possibilitar acesso gratuito e por envolver uma dinâmica focada no combate, onde a ação acontece continuamente, diferentemente de

alguns MOBAs. Segue-se uma imagem referente ao jogo (Figura 11).

Figura 11 – Imagem retirada do jogo TOME: Immortal Arena.



Fonte: O autor.

Strife: Desenvolvido pela S2 Games e lançado em 22 de maio de 2015, este jogo *free-to-play* tem como objetivo principal o trabalho em equipe e ação constante. Tendo o *design* voltado para esses aspectos, o jogo se torna útil para a lista devido ao foco do *design* e na fluidez do combate. Segue-se uma imagem referente ao jogo (Figura 12).

Figura 12 – Imagem retirada do jogo Strife.



Fonte: O autor.

Arena of Fate: Desenvolvido pela Crytek e anunciado em maio de 2014, esse jogo traz personagens dos contos de fantasia assim como personagens históricos. O jogo encontra-se nessa lista por partir de uma mecânica única, onde a defesa de sua base é feita separadamente do ataque a base inimiga. Segue-se uma imagem referente ao jogo (Figura 13).

Figura 13 – Imagem retirada do jogo Arena of fate.



Fonte: O autor.

Orcs Must Die! Unchained: Desenvolvido pela Robot Entertainment e lançado em 29 de março de 2016. Esse jogo está presente pois traz a união de gêneros diferentes, sendo eles: jogos de defesa de torre e **MOBAs**, a sua similaridade com jogos de defesa de torre e o posicionamento da câmera em terceira pessoa tendo o *mouse* como "mira" faz com que esse jogo se torne uma nova fonte para identificação de novos padrões. Segue-se uma imagem referente ao jogo (Figura 14).

Figura 14 – Imagem retirada do jogo Smite Orcs Must Die! Unchained.



Fonte: O autor.

Guardians of Middle-Earth: Desenvolvido pela Monolith Productions e Zombie Studios, publicado pela Warner Bros. Interactive Entertainment e lançado para PC em 29 de agosto de 2013. Esse jogo se baseia no universo fictício de "Senhor dos Anéis" e tem como foco principal a cooperação entre os jogadores de um mesmo time. O jogo também enfatiza

o uso não somente em computadores, mas também em consoles (Microsoft XBOX360 e Sony PlayStation 3), visando uma experiência mais cinematográfica ao jogador. Segue-se uma imagem referente ao jogo (Figura 15).

Figura 15 – Imagem retirada do jogo Guardians of Middle-Earth.



Fonte: O autor.

Supernova: Desenvolvido pela Primal Games, publicado pela Bandai Namco o jogo se apresenta atualmente em *open-beta*. Esse jogo em estilo *sci-fi* apresenta novas mecânicas que ajudam o jogador dentro de uma partida, assim como a possibilidade dos próprios jogadores para criarem tropas controladas pelo computador para ajudarem na conquista da base inimiga. Segue-se uma imagem referente ao jogo (Figura 16).

Figura 16 – Imagem retirada do jogo Supernova.



Fonte: O autor.

4.1.2 Lista de jogos pelo gênero FPSs

DOOM: Desenvolvido pela ID Software e lançado em 1993, este jogo foi um dos pioneiro do gênero, e que apresentou inovações tanto na parte gráfica quanto na forma com que o jogo interagia com o jogador, como por exemplo o modo como era mostrado a vida do personagem. Partindo desse ponto, o jogo se torna uma ótima fonte para o descobrimento e evolução de padrões, além de ser usado como referência para muitos outros que vieram depois deste. Segue-se uma imagem referente ao jogo (Figura 17).

Figura 17 – Imagem retirada do jogo Doom.



Fonte: O autor.

DOOM 2: Desenvolvido pela ID Software e lançado em 1994. É o segundo título da franquia DOOM, que devido a seu sucesso e popularidade, esse jogo alcançou diversas plataformas. Com a abertura de seu código-fonte posteriormente, a facilidade de expandir o jogo para outras plataformas aumentou, alcançando até mesmo dispositivos móveis. Seu desenvolvimento gráfico é muito parecido com o primeiro jogo, mas se destaca na elaboração de níveis e na adoção de padrões não-lineares de mapas. Segue-se uma imagem referente ao jogo (Figura 18).

Figura 18 – Imagem retirada do jogo Doom 2.



Fonte: O autor.

DOOM 3: Desenvolvido pela ID Software, distribuído pela Activision e lançado em agosto de 2003, este jogo ganhou vários prêmios na Electronic Entertainment Expo (E3), feira internacional dedicada a jogos, além de ser o terceiro jogo da franquia. O jogo apresentou mecânicas inovadoras e formas novas de interação com o usuário, tendo lugar nessa lista como uma boa forma de mostrar a evolução tanto de elementos gráficos como de formas de interação. Segue-se uma imagem referente ao jogo (Figura 19).

Figura 19 – Imagem retirada do jogo Doom 3.



Fonte: O autor.

Counter-Strike 1.6: Desenvolvido pela Valve e lançado em junho de 1999, esse FPSs abriu muitas portas, disponibilizando uma plataforma online se tornando muito popular na época, ligas profissionais viram neste jogo uma possibilidade de ganho monetário, levando o jogo a campeonatos, onde os ganhadores eram recompensados. A popularidade

torna uma ótima fonte de embasamento, já que por este, foram inspirados muitos outros jogos, possibilitando o estudo de padrões. Uma imagem do jogo pode ser observada na Figura 20.

Figura 20 – Imagem retirada do jogo Counter-Strike 1.6.



Fonte: O autor.

Battlefield 3: Desenvolvido pela Electronic Arts e lançado em outubro de 2011, vendeu mais de cinco milhões de jogos e recebeu muitos elogios da crítica, tornando-se muito popular e uma boa base para o estudo de boas técnicas de usabilidade. Segue-se uma imagem referente ao jogo (Figura 21).

Figura 21 – Imagem retirada do jogo Battlefield 3.

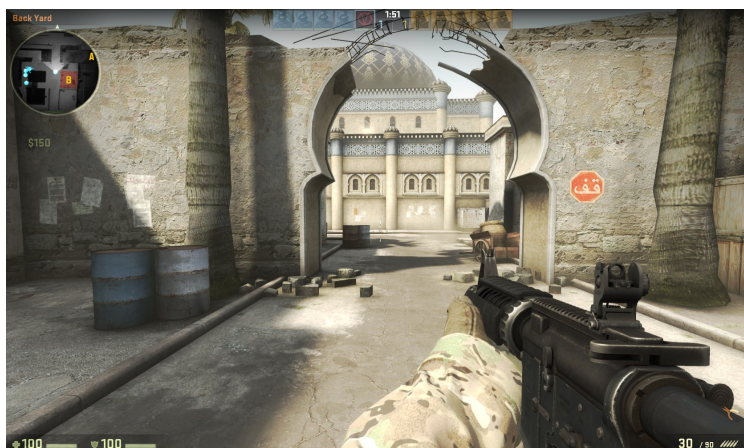


Fonte: O autor.

Counter-Strike: Global Offensive: Desenvolvido pela Valve e lançado em Agosto de 2012, este dando sequência ao sucesso alcançado por outro jogo da franquia. Sua

popularidade ganhou cenário competitivo, onde times de todo mundo jogam uns contra os outros por prêmios que podem chegar a um milhão de dólares, o que torna esse jogo viável para a prática de descobrimento de padrões dado seu embasamento pelo público desde jogos anteriores da franquia. Segue-se uma imagem referente ao jogo (Figura 22).

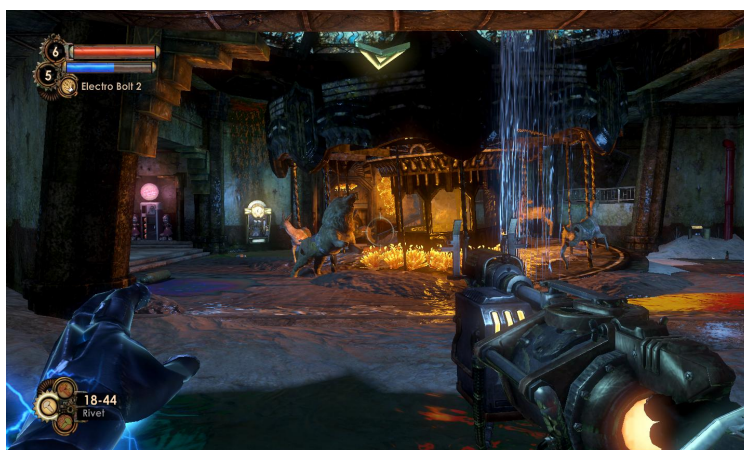
Figura 22 – Imagem retirada do jogo Counter-Strike: Global Offensive.



Fonte: O autor.

Bioshock: Desenvolvido pela 2K Boston e 2K Australia e lançado em agosto de 2007, ganhou o apreço do público por introduzir histórias irresistíveis, juntamente com elementos gráficos muito atraentes. Esse jogo tem potencial para ser uma ótima fonte de estudo sobre padrões, já que se utiliza de elementos dinâmicos, bem como estudo de design simplicistas e atraentes, deixando o jogo mais dinâmico e intuitivo possível. Segue-se uma imagem referente ao jogo (Figura 23).

Figura 23 – Imagem retirada do jogo Bioshock.



Fonte: O autor.

Bioshock Infinite: Desenvolvido pela Irrational Games e lançado em março de 2013. Terceiro jogo da franquia, apresentando história irresistível, juntamente com elementos gráficos muito atraentes. Esse jogo utiliza mecânicas dos jogos anteriores assim como elementos para tornar o jogo ainda mais dinâmico e intuitivo possível. Segue-se uma imagem referente ao jogo (Figura 24).

Figura 24 – Imagem retirada do jogo Bioshock Infinite.



Fonte: O autor.

Left 4 Dead: Desenvolvido pela Turtle Rock Studios, atual Valve, foi lançado em novembro de 2008. Apresentando ação frenética e a necessidade de cooperação em equipe. Este jogo foi selecionado pois seu objetivo final torna-se muito similar aos jogos [MOBA](#). Segue-se uma imagem referente ao jogo (Figura 25).

Figura 25 – Imagem retirada do jogo Left 4 Dead.



Fonte: O autor.

Dukem Nukem 3D: Desenvolvido pela 3D Realms, publicado pela GT Interactive

e lançado em maio de 1996. Terceiro jogo da franquia, apresentando história irresistível, juntamente com ação frenética. Esse jogo utiliza elementos gráficos muito similares a outros jogos apresentados na lista. O jogo foi selecionado a fim de comparação da disponibilidade de mesmos padrões em diferentes jogos. Segue-se uma imagem referente ao jogo (Figura 26).

Figura 26 – Imagem retirada do jogo Duke Nukem: 3D.



Fonte: O autor.

F.E.A.R.: Desenvolvido pela Monolith e lançado em 18 de outubro de 2005 para computadores. Um jogo de terror/horror apresenta uma abordagem focada em *jump scares*, onde a música e o susto vem através de rápida mudança no tela. Esse jogo foi selecionado a fim de investigar padrões envolvendo o gênero de terror em FPS e a existência desses em outros jogos dessa lista. Segue-se uma imagem referente ao jogo (Figura 27).

Figura 27 – Imagem retirada do jogo F.E.A.R.



Fonte: O autor.

Far Cry 3: Desenvolvido pela Ubisoft e lançado em 4 de dezembro de 2012. Esse jogo contém personagens icônicos e uma trama muito bem desenvolvida, prendendo o jogador ao decorrer da história. O jogo foi selecionado a fim de investigar padrões envolvendo a imersão do jogador com a interface disponibilizada. Segue-se uma imagem referente ao jogo (Figura 28).

Figura 28 – Imagem retirada do jogo Far Cry 3.



Fonte: O autor.

4.2 Gabarito para catalogar padrões

A criação do gabarito apresentado nesse trabalho teve como base um apresentado por [Folmer \(2006\)](#), utilizado em conjunto com outro apresentado por [Kreimeier \(2002\)](#), além de mostrar se este padrões já pertence a algum estudo ou trabalho. O gabarito para catalogar os padrões segundo esse trabalho, apresenta os seguintes itens:

- Nome: O nome representa um evocativo da resolução do problema em questão, ao momear um padrão, deve-se poder imaginar do que se trata, em um nível maior de abstração;
- Outras Menções: A origem do respectivo padrão, se foi encontrado pelo autor em determinado trabalho, fórum, comunidade de desenvolvimento, etc;
- Contexto e Problema: Uma descrição detalhada ajuda a demarcar os objetivos do padrão, bem como suas limitações;
- Solução: Proporciona uma solução satisfazível para o problema em questão, levando em consideração o seu custo/benefício;

- Princípio Heurístico: Deve remeter às heurísticas de usabilidade de [Nielsen \(1994\)](#) relacionadas ao problema que o padrão resolve, onde uma solução satisfazível para um problema genérico pode ser reduzida para uma específica para jogos;
- Exemplos: Apresenta exemplos de jogos que utilizam o padrão proposto, e que, tiveram o problema mencionado, devidamente solucionado;

4.3 Identificação de padrões já catalogados

A forma usada para recuperar padrões para jogos já catalogados foi através de pesquisas utilizando o Google Acadêmico, pesquisas em fóruns de base tecnológica e do site da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior ([CAPES](#)). A partir desses foram coletados padrões que foram reescritos conforme o gabarito apresentado neste trabalho. A seguir, são mostrados alguns desses padrões apresentados por [Kreimeier \(2002\)](#) que encontram-se dentro do escopo desse trabalho.

Em seu trabalho, [Kreimeier \(2002\)](#) apresenta uma lista de padrões para jogos. O modelo utilizado por ele consiste em quatro elementos essenciais:

- Nome: Descreve o padrão, em um nível maior de abstração;
- Problema: Descreve o problema, por meio deste deve-se chegar ao objetivo a ser atingido;
- Solução: Descreve os mecanismos e as entidades que serão usadas para sanar o problema, não definindo um *design* em particular ou implementação concreta;
- Consequências: Descreve as possíveis consequências geradas pela utilização da solução abordada, levando em conta tanto o custo-benefício, como a criação de novos problemas menores;

O primeiro padrão a ser apresentado será o "Weenie Chain". Esse padrão descreve a possível perda do senso direcional do jogador, dada a inexistência de indicadores para onde se deve prosseguir. A solução apontada por Kreimeier visa criar uma corrente de instruções perceptíveis, para guiar o jogador do início ao fim do jogo. Em contra partida, uma sequência de passos pode, de alguma maneira, limitar a experiência do usuário, tornando o jogo desagradável. A necessidade de utilização de informações que sejam úteis, e ao mesmo tempo superficiais, diminui o risco de impacto da consequência mencionada.

A tabela 1 mostra a disponibilidade desse padrão em alguns jogos utilizados neste trabalho.

Tabela 1 – Padrão Weenie Chain.

Jogo	Apresenta o Padrão?	Como é apresentado?
League of Legends	Sim	Apresenta opção que habilita dicas de como deve-se prosseguir enquanto o jogo é praticado.
Heroes of Newerth	Sim	Apresenta opção que habilita dicas de como deve-se prosseguir enquanto o jogo é praticado.
Heroes of the Storm	Sim	Apresenta opção que habilita dicas de como deve-se prosseguir enquanto o jogo é praticado.
Smite	Sim	Apresenta vídeos explicativos dentro do jogo, explicando e orientando o jogador do como se jogar, além de introduzi-lo aos objetivos principais e dicas de como utilizar cada campeão.
Battlefield 3	Sim	Ao iniciar o jogo, o jogador é levado a um tutorial, onde são explicadas todas as mecânicas, bem como como se dará a procedência do jogo.
Bioshock Infinity	Sim	Ao iniciar o jogo, o jogador é levado a um tutorial, onde são explicadas todas as mecânicas, bem como como se dará a procedência do jogo.

Outro padrão apresentado por Kreimeier é o "Privileged Move". Este descreve a impossibilidade de uma entidade interferir com a ação de outras, ou a proibição de outras entidades interferirem na ação do jogador. A solução encontrada pelo autor foi a adição de movimentos exclusivos, dando privilégio na ação a entidade desejada, além da proteção da entidade em questão. Em contra partida, essa prioridade pode gerar desconforto para o jogador, restringindo seus movimentos ou o induzindo a tomar certas decisões.

A tabela 2 mostra a disponibilidade desse padrão em alguns jogos que serão utilizados neste trabalho.

Tabela 2 – Padrão Privileged Move.

Jogo	Apresenta o Padrão?	Como é apresentado?
League of Legends	Sim	Certos campeões têm privilégio na utilização de suas habilidades, levando-os a se tornarem indestrutíveis por certo período de tempo.
Heroes of Newerth	Sim	Certos campeões têm privilégio na utilização de suas habilidades, levando-os a se tornarem indestrutíveis por certo período de tempo.
Heroes of the Storm	Sim	Certos campeões têm privilégio na utilização de suas habilidades, levando-os a se tornarem indestrutíveis por certo período de tempo.
Smite	Sim	Certos campeões têm privilégio na utilização de suas habilidades, levando-os a se tornarem indestrutíveis por certo período de tempo.
DOOM	Sim	A travessia de monstros por poças de ácido para alcançar o jogadores, prática impossível para o jogador, já que essas poças matam-no.
Counter-Strike 1.6	Sim	Período de invulnerabilidade do jogadores por breves segundos enquanto compra-se equipamentos.
Bioshock Infinity	Sim	Quando os inimigos especiais são apresentados para o jogador, eles são indestrutíveis.

A lista completa de padrões descritos por [Kreimeier \(2002\)](#) pode ser encontrada em seu trabalho. A utilização desses padrões no presente trabalho é destacada devido a maior ocorrência destes nos gêneros estudados.

4.4 Identificação de padrões ainda não catalogados

A identificação de novos padrões se deu avaliando problemas de usabilidade. A utilização de avaliações heurísticas usando os princípios de [Nielsen \(1994\)](#) em conjunto com a revisão de problemas apresentados em jogos mais antigos e corrigidos em versões mais recentes apresenta resultados relevantes que, dependendo de sua natureza, podem vir a tornar-se padrões.

Esta avaliação dos jogos pelo autor desse trabalho consiste num processo de duas etapas. Na primeira, cada heurística foi estudada e, ao jogar cada um dos jogos, verificou-se a existência das heurísticas tanto em elementos de interface quanto métodos de interação, sejam elas via teclado, *mouse* ou efeitos visuais. A cada nova interação descoberta, refazia-se a busca na lista de jogos para observar se essa interação ocorria nos demais jogos.

Um exemplo da aplicação dessa técnica pode ser percebido no padrão “Múltiplos Indicadores de Vida”. Ao perceber a importância do indicador de vida dentro do jogo “League of Legends”, nota-se a existência de barras alternativas para manter esse recurso visível, percorreu-se a lista de jogos para checar se isso acontecia nos demais jogos. Assim,

os demais jogos da lista foram verificados em busca desse mesmo padrão. A Figura 29 mostra, com a ajuda de retângulos brancos, essa ocorrência dentro dos jogos “League of Legends” no “canto superior esquerdo”, “Defense of the Ancients” no “canto superior direito”, “Arena of Fate” no “canto inferior esquerdo” e “Doom 2” no “canto inferior direito”.

Figura 29 – Imagem referente a existência de vários indicadores de vida em múltiplos jogos.



Fonte: O autor.

Na segunda etapa, tentava-se retirar elementos da interface dentro dos jogos e observar o impacto que elas tinham sobre o jogo. Se alguma ação fosse necessária para suprir a falta do elemento, observava-se nos outros jogos como eles a cobriam, ou se utilizavam o mesmo tipo de solução.

Um exemplo dessa aplicação pode ser percebido no padrão "Visibilidade de Mapa Ampliada". Em jogos de equipe como os jogos [MOBA](#) e jogos [FPS](#), saber a posição de seus aliados e inimigos é de suma importância para chegar a vitória. Se o jogador depender somente de seu campo de visão sem ter qualquer informação sobre seus arredores ou até mesmo o mapa do jogo, isso o torna um alvo fácil, além de impedir uma melhor comunicação com seus companheiros. A Figura 30 mostra os jogos "Counter-Strike; Global Offensive" no canto superior esquerdo, "Heroes of Newerth" no canto superior direito, "FarCry 3" no canto inferior esquerdo e "Defense of the Ancients 2" no canto inferior direito lidando com esse problema através de um minimapa.

Figura 30 – Imagem referente a existência de um minimapa em múltiplos jogos.



Fonte: O autor.

Padrões encontrados nessa avaliação foram devidamente catalogados. Já padrões identificados por meio de fóruns (não catalogados apropriadamente) são verificados nos jogos da lista e se plausíveis, adequá-los ao gabarito. Esse catálogo se deu a partir de um gabarito em um documento Word Excel, sendo disponibilizado no Google Drive, contendo a lista completa de padrões e suas respectivas disponibilidades nos jogos estudados.

5 Resultados

Os resultados apresentados nesta seção são divididos em padrões refatorados, que já foram catalogados de alguma forma, e padrões propostos, que foram encontrados pelo autor ou que não foram propriamente catalogados.

Para padrões estudados são apresentado a adaptação dos padrões para o gabarito apresentado nesse trabalho, assim como os resultados encontrados ao analisá-los nos jogos propostos.

Para padrões propostos pelo autor são apresentados os resultados ao analisar os jogos listados e a aplicação dos supostos padrões dentro do gabarito.

5.1 Padrões Refatorados

É mostrado dois exemplos de padrões refatorados dentro do gabarito desse trabalho. Além de mostrar as mudanças envolvendo o gabarito utilizado nesse trabalho e o utilizado pelo autor que o identificou.

5.1.1 Padrão Priorização de Ações Específicas

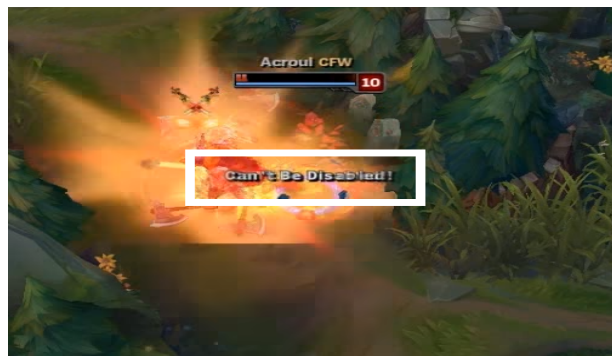
Para padrões já catalogados, um exemplo de padrão remete-se a [Kreimeier \(2002\)](#). A mudança no método de catalogação trouxe algumas vantagens. Ao utilizar a padronização do presente trabalho deixa-se mais claro o contexto em que o problema está envolvido dentro de um jogo, além de atrelá-lo a um princípio heurístico, bem como a apresentação de exemplos e imagens do padrão em jogos. A adaptação do padrão “Movimento Prioritário” é um exemplo remetendo à [Kreimeier \(2002\)](#). Segundo esse padrão, elementos dentro de um jogo podem se sobresscrever e o jogo deve lidar com isso de forma adequada.

- Nome: Priorização de Ações Específicas
- Outras Menções: [Kreimeier \(2002\)](#) (Movimento prioritário)
- Contexto e Problema: Às vezes, elementos de interface, entidades e locais não podem ou necessitam de permissão para serem acessados ou interagirem, tendo em vista a necessidade de priorizar a movimentação ou ação de determinado objeto, a interferência com este deve ser regrada e protegida, dada sua interferência com outros objetos. Em jogos [MOBA](#), isso acontece quando um personagem, por exemplo, usa uma habilidade que o deixa imune a certas mecânicas, como a perda de vida e se outro personagem lhe causa dano, o jogo deve identificar a ação prioritária

e exercê-la. Já em jogos **FPS**, áreas do mapa podem não ser acessíveis até que o jogador as desbloqueie, completando certas missões e objetivos. Outro exemplo, remete a aparição de inimigos especiais que durante um curto período de tempo, são invencíveis até que sua apresentação seja realizada. Ou até mesmo, em um jogo **FPS**, o período de compra de equipamento ao iniciar uma partida.

- Solução: Para jogos **MOBA**, introduzir mecânicas regradas dentro do ambiente do jogo para que quando houver sobreposição de informação, o jogo saiba qual é a ação que deve ser priorizada e mostrar ao usuário do porque dessa priorização, seja por uma mensagem de *feedback* dizendo que o inimigo não pode ser silenciado ou que não pode tomar dano. Já para jogos **FPS**, quando uma área somente é acessada após o término de alguma missão ou objetivo, o ambiente do jogo pode ser dividido em áreas distintas usando portas fechadas, portões e até mesmo magia. Outra solução, essa visando tanto a apresentação de inimigos quanto a fase de compra de armas ao iniciar um jogo, seria impedir o personagem ou inimigo de sofrer dano, aumentando assim sua integridade na hora do combate.
- Principio Heurístico: Prevenção de erros, Flexibilidade e Eficiência de Uso, Visibilidade de Status do Sistema
- Exemplos: Defense of the Ancients, League of Legends, Heroes of the Newerth, Defense of the Ancients 2, Heroes of the Storm, Magika: Wizard Wars, Smite, TOME: Immortal Arena, Strife, Arena of Fate, Guardian of Middle-Earth, Supernova, Duken Nukem: 3D, DOOM, DOOM 2, DOOM 3, Counter-Strike 1.6, Counter-Strike: Global Offensive, Left4Dead, Far Cry 3, Battlefield 3, F.E.A.R., Bioshock, Bioshock Infinite

Figura 31 – Utilização do padrão “Movimento Prioritário” explicitada por “can’t be disabled”(Não pode ser impedido).



Fonte: O autor.

Ao estudar esse padrão, nota-se que para diferentes gêneros de jogos, os padrões podem ou não se encaixar. O padrão acima é muito presente em jogos com muitas interações entre diferentes personagens, o que o faz ser altamente utilizado em jogos **MOBA**. Esse gênero de jogo normalmente apresenta uma gama de interações muito mais diversificada que jogos **FPS**, onde muitas das interações se resumem a atirar com armas de fogo e ferir o inimigo.

5.1.2 Padrão Dificuldade de Nível Adaptativa

Outro padrão já catalogado, remete-se a [Folmer \(2006\)](#). Para esse método, as mudanças no método de catalogação foram mais sutis, dada a utilização desse como base para a criação do formato atual. Algumas mudanças incluem a remoção das forças envolvidas, dado o número repetitivo e resultado implícitos de forças, a união do contexto e do problema, explicando-o e apresentando-o em um mesmo local, a adição de um nome para evocar o problema e a união do problema e do porquê dentro da solução. Essa adaptação é o padrão "Dificuldade de nível adaptativa". Segundo esse padrão, o jogo deve sugerir ou mudar implicitamente a dificuldade do jogo baseado no desempenho do jogador.

- Nome: Dificuldade de nível adaptativa
- Outras Menções: [Folmer \(2006\)](#)
- Contexto e Problema: Em muitos jogos, principalmente aqueles com campanha *offline*, é normal oferecer diversos níveis de dificuldades para diferentes tipos de jogador, os termos mais comuns para níveis de dificuldade são fáceis, normais e difíceis. Dependendo da dificuldade, inimigos ficam mais fortes ou fracos, mais ou menos dicas, mais ou menos assistência às ações do jogador é fornecida. Normalmente, o jogador escolhe um nível de dificuldade ao iniciar o jogo, e geralmente não é possível mudar para um nível de dificuldade diferente no meio do caminho sem começar de novo, o que limita a liberdade do jogador e pode até, fazê-lo desistir do jogo.
- Solução: Várias abordagens podem ser implementadas, tais como a permissão para o jogador de troca de dificuldade dentro das opções do jogo, podendo ou não sugerir a diminuição da dificuldade atual de acordo com o desempenho do jogador. Por exemplo, o jogo poderia sugerir uma dificuldade mais fácil depois que o jogador falhou/tenha sido morto várias vezes para passar um ponto no jogo. Outra abordagem seria ajustar automaticamente a dificuldade com base no desempenho do jogador. Removendo assim os "níveis de dificuldade tradicionais". Por exemplo, toda vez que o jogador morre ou falha, existe a chance dele mudar para a próxima configuração

de dificuldade mais fácil ou se estiver jogando bem por um tempo, há uma chance de o jogador subir um nível de dificuldade

- Principio Heurístico: Flexibilidade e Eficiência de Uso
- Exemplos: Left4Dead, Far Cry 3, Battlefield 3, F.E.A.R., Bioshock, Bioshock Infinite

Figura 32 – Utilização do "Padrão Dificuldade de nível adaptativa" em Battlefield 3.



Fonte: O autor.

Observando esse padrão, nota-se que jogos *offline* tendem a ter opções mais adequadas de controle de dificuldade. Isso muitas vezes se torna difícil no nivelamento de jogos *online*, onde jogadores praticam com outros jogadores. Dadas as diferentes temáticas e objetivos, um exemplo disso é o padrão de usabilidade “Dificuldade de nível adaptativa” mostrado acima, proposto por [Folmer \(2006\)](#).

Dos jogos apresentados, em sua maioria os **MOBAs** se utilizam de técnicas como nível ou tempo de jogo para nivelar a jogabilidade dos jogadores, algo que muitas vezes não funciona. Já jogos com campanhas *offline* têm vantagem, dada a possibilidade de mudança dessa dificuldade a qualquer momento do jogo, somente entrando em um menu.

Os demais padrões encontrados pelo autor remetem a [Game Accessibility Guidelines \(2012\)](#) que, após identificados dentro dos jogos foram associados por terem um grau de similaridade ou o contexto englobado dentro de padrões já apresentados nesse catálogo. Esses padrões são apresentados abaixo dentro do formato utilizado pelo autor.

5.1.3 Padrão Tutoriais para Iniciantes

- Nome: Tutoriais para Iniciantes

- Outras Menções: [Kreimeier \(2002\)](#), [Game Accessibility Guidelines \(2012\)](#)
- Contexto e Problema: Iniciar um jogo sem nenhum tipo de ajuda pode se tornar algo estressante, ainda mais se o jogo for *online* e necessitar de outros jogadores que podem ou não já ter experiência. Sendo assim, o usuário pode não ter nenhuma noção de seus controles, objetivos e mecânicas e cabe ao jogo ensiná-lo. Jogos que apresentam algum tipo de tutorial tendem a melhorar a jogabilidade de novos jogadores, já que o conteúdo do jogo é introduzido por partes, dando tempo para o jogador se adaptar à sua maneira.
- Solução: Ao iniciar um jogo pela primeira vez deve-se perguntar se o novo usuário já tem alguma experiência com esse tipo ou gênero de jogo e se quer participar de um tutorial interativo que irá lhe mostrar através de um passo a passo o que deve ser feito. Esse tutorial pode ocorrer tanto para jogos [MOBA](#) quanto para jogos [FPS](#) e deve introduzir periodicamente os controles esperando ou não uma resposta do jogador. Após apresentar os controles, o tutorial deve introduzir ao jogador qual seu objetivo principal e, se houver, objetivos secundários, assim como deixá-lo a par de mecânicas-chave que podem acontecer durante uma partida.
- Princípio Heurístico: Reconhecimento em vez de memorização.
- Exemplos: League of Legends, Defense of the Ancients 2, Heroes of the Storm, Smite.

Figura 33 – Utilização de “Tutoriais para Iniciantes” em Defense of the Ancients 2.

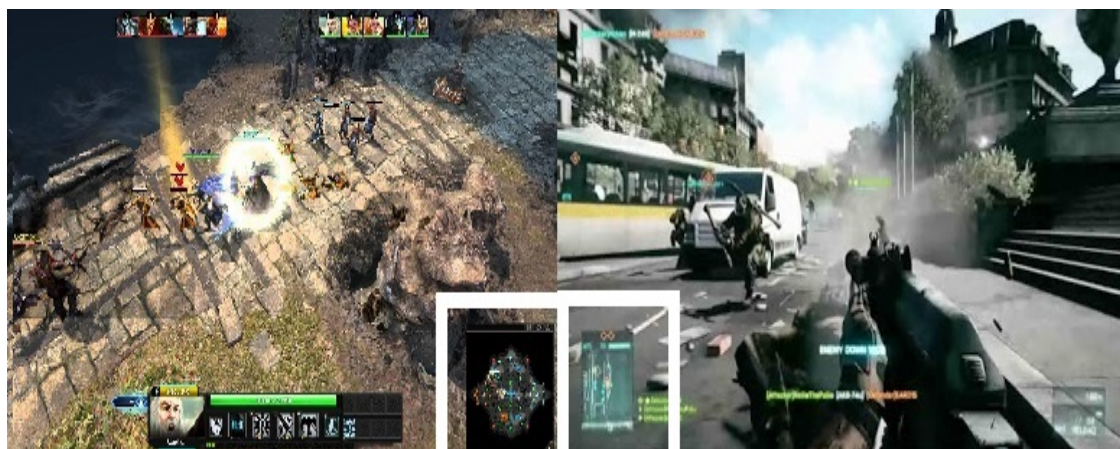


Fonte: O autor.

5.1.4 Padrão Visibilidade de Mapa Ampliada

- Nome: Visibilidade de Mapa Ampliada
- Outras Menções: [Kreimeier \(2002\)](#), [Game Accessibility Guidelines \(2012\)](#)
- Contexto e Problema: Durante o jogo, a área observada pelo jogador limita-se à visão de seu personagem em jogos [FPS](#) e para jogos [MOBA](#) uma visão de seu personagem além de uma pequena área ao redor dele, forçando-o a depender de outros recursos para uma percepção mais abrangente do que acontece pelo mapa inteiro. Ter que depender somente dessa pequena área em jogos [FPS](#) dificulta ao jogador saber o que está acontecendo com seus aliados e seus objetivos. Já para jogos [MOBA](#), ter que mover o campo de visão de seu personagem para percorrer outra região do mapa o torna vulnerável a ataques de inimigos, além de não cobrir uma área mais abrangente.
- Solução: Adicionar à interface do jogador uma área com uma visão mais abrangente, não limitando-o a visão da câmera de seu personagem. Para jogos [FPS](#), o uso de um radar em algum dos cantos da tela pode ajudar na percepção de informações, que colaboram com: trabalho em equipe, posicionamento de aliados, investida inimiga e proteção ou ataque a objetivos específicos. Já para jogos [MOBA](#), um minimapa em algum dos cantos da tela pode prover informações necessárias para a tomada de decisões em meio ao jogo, fazendo com que uma investida inimiga em outro lugar do mapa faça-o ajudar seus aliados ou tomar outros objetivos, como torres e roubar recursos sem apresentar muito perigo.
- Princípio Heurístico: Flexibilidade e Eficiência de Uso, Visibilidade de Status do Sistema
- Exemplos: Defense of the Ancients, League of Legends, Heroes of the Newerth, Defense of the Ancients 2, Heroes of the Storm, Magika: Wizard Wars, Smite, TOME: Immortal Arena, Strife, Arena of Fate, Orcs Must Die! Unchained, Guardian of Middle-Earth, Supernova, Counter-Strike 1.6, Counter-Strike: Global Offensive, Far Cry 3, Battlefield 3

Figura 34 – Utilização de “Visibilidade de Mapa Ampliadas” em Guardian of Middle-Earth e Battlefield 3.



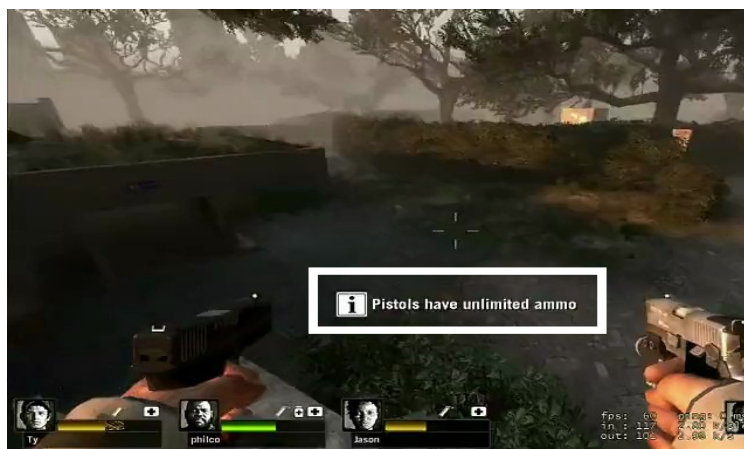
Fonte: O autor.

5.1.5 Padrão Uso de Linguagem Limpa

- Nome: Uso de Linguagem Limpa
- Outras Menções: [Game Accessibility Guidelines \(2012\)](#)
- Contexto e Problema: Ao decorrer de um jogo, vários diálogos são apresentados ao jogador e dependendo de sua escrita podem fazer o jogador mudar de decisão. Em jogos **MOBA** e jogos **FPS online**, esse padrão é encontrado de forma minimizada, já que normalmente as informações sobre uma partida, itens, objetivos e inimigos são, na maioria dos casos, constantes. Agora para jogos **FPS** com campanhas *offline*, esse padrão se torna de suma importância, já que ao longo do jogo, informações são constantemente apresentadas ao jogador, sendo essas: como manejar uma arma recém adquirida, objetivos principais e objetivos secundários. A interpretação da nova informação deve ser clara, não deixando espaço para incertezas.
- Solução: Para todos os jogos, o uso de uma linguagem mais clara, sem palavras rebuscadas e contendo um alto grau de objetividade. Por exemplo, em vez de dizer “essa arma tem munição ilimitada”, o jogo diz “pistolas têm munição infinita” para indicar claramente o tipo de arma sem depender do jogador perceber que ele está atualmente usando as pistolas. Outro exemplo encontra-se na forma limpa de escrita, em vez de descrever uma ação meticulosamente, focar-se no objetivo principal e escrever o mínimo e o necessário para a compreensão da frase, um exemplo disso seria o uso de expressões como: "Clique aqui para salvar" ao invés de "Clicando no botão abaixo, seu processo até o momento, bem como as preferencias de seu personagem, serão gravados. "

- Princípio Heurístico: Visibilidade de Status do Sistema, Consistência, Estética e design minimalista
- Exemplos: Defense of the Ancients, League of Legends, Heroes of the Newerth, Defense of the Ancients 2, Heroes of the Storm, Magika: Wizard Wars, Smite, TOME: Immortal Arena, Strife, Arena of Fate, Orcs Must Die! Unchained, Guardian of Middle-Earth, Supernova, Counter-Strike: Global Offensive, Left4Dead, Far Cry 3, Battlefield 3, F.E.A.R., Bioshock, Bioshock Infinite.

Figura 35 – Utilização de “Linguagem Limpa” em Left4Dead.



Fonte: O autor.

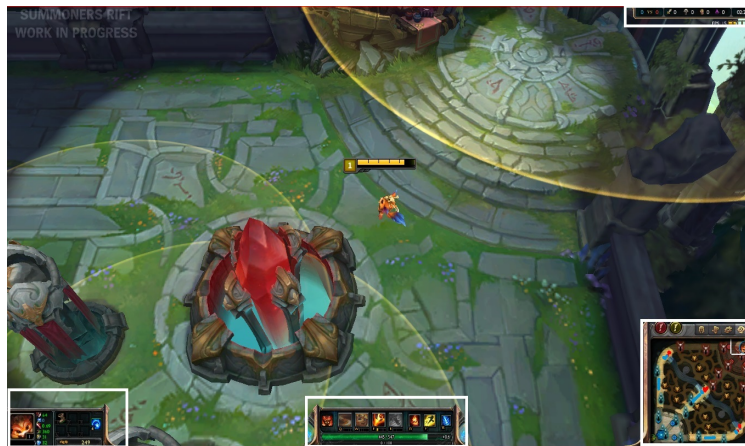
5.1.6 Padrão Uso de HUD Personalizável

- Nome: HUD¹ Personalizável
- Outras Menções: [Game Accessibility Guidelines \(2012\)](#)
- Contexto e Problema: O campo de visão do jogador é limitado pelo tamanho da tela, sendo esse espaço dividido entre o campo de visão de seu personagem e a HUD, que seria a interface de ajuda e alerta do jogador. Tornar a HUD muito grande melhora o entendimento de informação relacionada ao personagem dentro do jogo, mas limita seu campo de visão. Por outro lado, uma HUD muito pequena aumenta o campo de visão, mas torna difícil o raciocínio perante o personagem. Existe ainda, a questão de HUD fixa, limitando as opções do jogador em tornar a interface mais agradável ao seu estilo de jogo, não disponibilizando ao jogador a troca de lugar dos elementos dentro da HUD.

¹ Do inglês *Heads-up Display*, é a interface de ajuda ao jogador

- Solução: Apresentar nas opções do jogo uma área específica para jogador realocar e remanejar a HUD como bem quiserem. Tornar essa interface maior ou menor deve vir do jogador para que o jogo se torna agradável para ele. Outra opção está na adição da Lei de Fitts, que é um modelo do movimento humano, predizendo o tempo necessário para mover-se, por exemplo com o *mouse*, de um ponto A da tela até um ponto B e usar isso ao criar ou realocar a HUD do jogador pode aumentar sua satisfação.
- Princípio Heurístico: Visibilidade de Status do Sistema, Estética e design minimalista, Flexibilidade e eficiência de uso.
- Exemplos: League of Legends, Heroes of the Newerth, Defense of the Ancients 2, Heroes of the Storm, Magika: Wizard Wars, Smite, Strife, Guardian of Middle-Earth, Supernova, Counter-Strike: Global Offensive, Left4Dead, Far Cry 3, Battlefield 3, Bioshock Infinite.

Figura 36 – Utilização de “Uso de HUD Personalizável” em League of Legends.



Fonte: O autor.

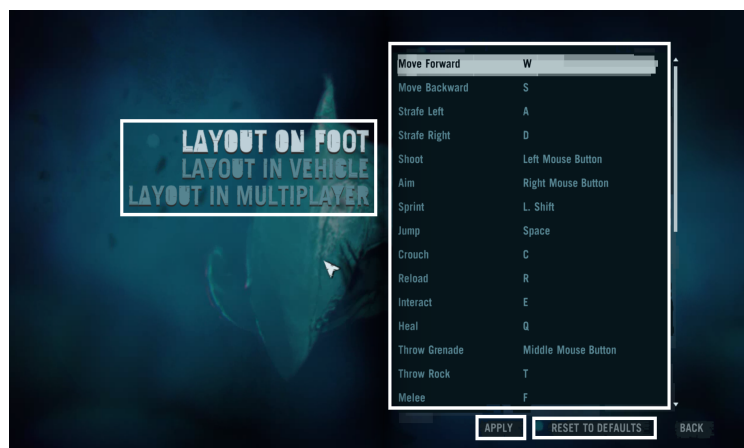
5.1.7 Padrão Teclas de Atalho Customizáveis

- Nome: Teclas de Atalho Customizáveis
- Outras Menções: [Game Accessibility Guidelines \(2012\)](#)
- Contexto e Problema: Muitos jogadores não se limitam às teclas de atalho que são dadas ao começarem a jogar, já que tanto para iniciantes como para veteranos, a realocação das teclas correspondentes às ações de um personagem podem não ser otimizadas. Privar um jogador a usar teclas personalizadas pode se tornar um

problema, uma vez que o jogador pode se sentir desconfortável e não se adaptar a restrição imposta a ele.

- Solução: Apresentar dentro das opções do jogo uma interface possibilitando a customização das teclas de atalho, não interferindo assim, em sua jogabilidade. Outro exemplo consiste também na rápida troca de armas em jogos FPS, onde ao tentar trocar para outra arma, em vez de um menu que mostra todas as opções de armas disponíveis para escolha, a arma é trocada instantaneamente por meio de uma tecla (ou combinação de teclas) de atalho. Esse exemplo acontece de forma diferente, por exemplo, em jogos MOBA, onde é possível o uso automático de habilidades utilizando o cursor como alvo ao pressionar a tecla referente a habilidade, ao invés de criar uma área tendo o *mouse* como ponto de referencia.
- Princípio Heurístico: Liberdade de controle fácil pro usuário, Flexibilidade e eficiência de uso.
- Exemplos: Defense of the Ancients, League of Legends, Heroes of the Newerth, Defense of the Ancients 2, Heroes of the Storm, Magika: Wizard Wars, Smite, TOME: Immortal Arena, Strife, Arena of Fate, Orcs Must Die! Unchained, Guardian of Middle-Earth, Supernova, Duken Nukem: 3D, DOOM, DOOM 2, DOOM 3, Quake 3, Counter-Strike 1.6, Counter-Strike: Global Offensive, Left4Dead, Far Cry 3, Battlefield 3, F.E.A.R., Bioshock, Bioshock Infinite.

Figura 37 – Utilização de “Teclas de Atalho Customizáveis” em Far Cry 3.



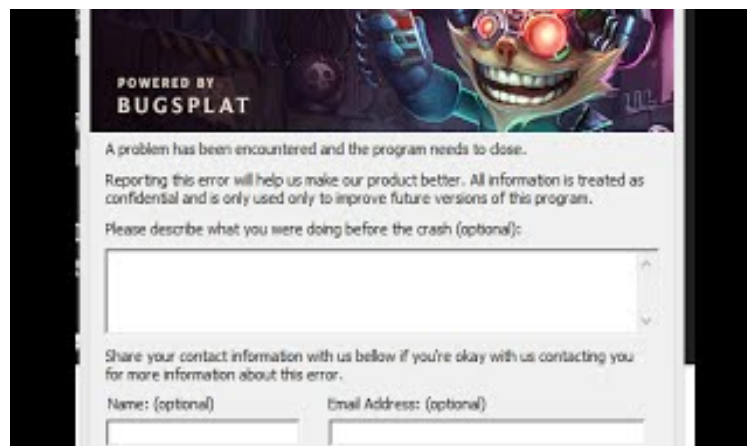
Fonte: O autor.

5.1.8 Padrão Uso de Alerta de Erro/Bug

- Nome: Alerta de Erro/Bug

- Outras Menções: [Game Accessibility Guidelines \(2012\)](#)
- Contexto e Problema: A qualquer momento do jogo, pode ocorrer um erro que seja, ou não, solucionável. O sistema deve se comportar de acordo e promover ao jogador algum tipo de *feedback* para sua melhor compreensão do que ocorreu. Esse *feedback* deve descrever da melhor maneira possível o que aconteceu com o jogo. Outra parte do padrão, envolve *bugs* gráficos, onde em jogos [MOBA](#) e [FPS](#), um personagem, podendo ser aliado, inimigo ou seu próprio, encontra-se “bugado²”, podendo estar preso em uma posição fixa no mapa, não podendo se mexer, ou estar desconfigurado visualmente.
- Solução: Caso o erro não seja solucionável, mostrar alguma mensagem de erro/*bug* ao jogador, informando a ocorrência desse problema, além de poder pedir ao jogador um relato do que aconteceu, enviando esta mensagem a empresa desenvolvedora do jogo, para que uma solução seja efetuada o mais rápido possível. Já se o erro/*bug* for solucionável, apresentar ao jogador possíveis soluções onde ele mesmo possa se recuperar ainda dentro do jogo, proporcionando assim uma melhor interação entre o jogador e o jogo.
- Princípio Heurístico: Ajude os usuários a reconhecerem, diagnosticarem e recuperarem-se de erros.
- Exemplos: League of Legends, Defense of the Ancients 2, Heroes of the Storm, Strife, Orcs Must Die! Unchained, Battlefield 3.

Figura 38 – Utilização de “Alerta de Erro/Bug” em League of Legends.



Fonte: O autor.

² Jargão usado por jogadores, podendo expressar a incapacidade de mover-se, ou impossibilitado de realizar ações

5.2 Padrões Identificados

Nessa seção são apresentados os padrões identificados ao aplicar as avaliações heurísticas, além de conter explicações ao decorrer dessas avaliações.

5.2.1 Padrão Verificação de Desempenho de Jogador

Um exemplo de padrão proposto apresentado no catálogo é o "Verificação de Desempenho de Jogador". Nele, desempenho e outras informações sobre a partida são apresentados para os jogadores facilitando a sua tomada de decisão durante uma partida ou a análise do um jogo após seu término. Segue abaixo o respectivo padrão:

- Nome: Verificação de Desempenho de Jogador
- Outras Menções: -
- Contexto e Problema: Durante e depois de partidas, mostrar o desempenho do jogador pode facilitar o entendimento de informações relevantes a ele. Ao analisar seus números, o jogador pode descobrir como melhorar sua jogabilidade, sua tomada de decisões e características implícitas de seu personagem ou outros personagens jogáveis.
- Solução: Essa abordagem pode ocorrer de duas formas, a primeira consiste na implementação de uma tabela *in-game*³ com o desempenho do jogador em conjunto com todos os outros jogadores ou “bots”⁴ que jogaram em conjunto com ele para determinada partida. Esse desempenho pode vir através de acurácia, KDA (do inglês Assassinos/Mortes/Assistências), número de tropas assassinadas, itens construídos e nível. A segunda forma, surge a partir da primeira, mostrando resultados mais concretos e específicos ao final do jogo. Dessa forma, o jogador pode analisar em números, seu dano causado a inimigos, o montando de dano que foi causado a ele, seu ouro obtido pelos abates de tropas inimigas, um gráfico condizente com a comparação do ouro ganho pelos dois times, além das informações já obtidas durante o jogo.
- Princípio Heurístico: Flexibilidade e eficiência de uso, Visibilidade de Status do Sistema.
- Exemplos: Defense of the Ancients, League of Legends, Heroes of the Newerth, Defense of the Ancients 2, Heroes of the Storm, Magika: Wizard Wars, Smite,

³ dentro do jogo

⁴ personagens controlados pelo próprio jogo

TOME: Immortal Arena, Strife, Arena of Fate, Guardian of Middle-Earth, Supernova, Counter-Strike 1.6, Counter-Strike: Global Offensive, Left4Dead, Battlefield 3.

Esse padrão foi encontrado ao sentir a necessidade de checar as estatísticas de todos os jogadores dentro de uma partida no jogo League of Legends. Para consolidar a ideia do padrão e adequá-lo aos moldes do gabarito, primeiramente foi indicado a qual heurística de usabilidade de Nielsen (1994) o padrão se referia, posteriormente qual o impacto que isso gerava ao jogar sem depender desse recurso, bem como a descrição do contexto onde o padrão é utilizado. Agora tendo um problema e sabendo que isso afeta sua jogabilidade, foi analisado dentro dos jogos da lista se o problema também existia ou, se não houvesse o problema, como o jogo o evitava. Após a observação notou-se que muitos jogos utilizavam ferramentas *in-game* para mostrar o desempenho de um ou mais jogadores na partida, proporcionando a ele dados para ver seu desempenho geral, ver o desempenho dos outros jogadores, compreender eventos que resultaram na vitória ou na derrota, e tomar melhores decisões de como agir nos próximos jogos.

Nesse jogo a tecla “TAB” por padrão abre uma janela contendo as estatísticas de cada jogador na partida, sendo essas o número de vezes que seu personagem foi morto, a quantidade de inimigos que eliminou, a quantidade de assistências na eliminação de inimigos por companheiros, tempo de objetivos secundários, itens de todos os jogadores. Toda essa informação é de suma importância para alguém que queira entender as fraquezas de sua equipe, entender as ameaças dos inimigos, criar estratégias e chegar a vitória, bem como analisar as forças de seu time e o quão bem eles e você estão indo ao decorrer de uma partida. Abaixo uma imagem ilustrativa do uso do padrão *in-game* (Figura 39).

Figura 39 – Tela de Resumo em League of Legends para Verificação de Desempenho de jogador.



Fonte: O autor.

Esse padrão também se fez presente após o término do jogo, apresentando gráficos mostrando o dano causado, a diferença entre o ouro das duas equipes, seus *status* pós-jogo, entre outros. Todas essas ferramentas juntas auxiliam o jogadores a se adaptar a situações diversas dentro de um jogo e melhorar a sua tomada de decisão para jogos futuros. Abaixo duas imagens representando o padrão em um pós-jogo (Figuras 40 e 41).

Figura 40 – Tela de Pós-jogo em League of Legends para Verificação de Desempenho de jogador.



Fonte: O autor.

Figura 41 – Tela de estatísticas em Pós-jogo de League of Legends para Verificação de Desempenho de jogador.

Overview	Grid	Graph						
Combat								
Champion Kills	0	5	3	2	5	3	5	
Deaths	2	5	4	3	0	3	3	
Assists	9	3	3	3	3	5	3	
Largest Killing Spree	0	3	0	0	5	2	3	
Largest Multi Kill	0	1	1	1	2	1	2	
Damage Done								
Damage Dealt to Champions	9,725	7,172	9,783	6,963	9,924	6,577	12,713	
Physical Damage to Champions	648	664	7,002	6,963	4,119	4,917	1,165	
Magic Damage to Champions	7,501	6,507	1,139	0	4,830	1,313	11,117	
Damage Dealt	40,603	58,746	85,597	108,356	80,382	75,659	80,916	
Physical Damage Dealt	2,551	13,652	74,934	108,356	44,118	46,324	11,515	
Magic Damage Dealt	33,716	45,094	8,793	0	29,313	22,131	68,970	
Largest Critical Strike	0	0	540	926	396	0	0	
Damage Taken & Healed								
Healing Done	60	480	4,039	205	600	970	404	
Damage Taken	6,436	11,096	18,246	9,864	11,982	15,419	11,040	
Physical Damage Taken	3,313	4,313	10,495	4,276	6,982	10,136	5,843	
Magic Damage Taken	2,905	6,466	7,697	5,398	5,000	4,888	4,400	
Misc.								
Gold Earned	7,378	7,754	7,867	8,475	7,958	6,960	7,352	
Turrets Destroyed	1	0	1	1	0	0	0	
Inhibitors Destroyed								

Fonte: O autor.

5.2.2 Padrão Múltiplos Indicadores de Vida

Outro exemplo de padrão proposto apresentado no catálogo é o "Múltiplos indicadores de Vida". Já que a vida (ou vitalidade) do personagem do jogador é, normalmente, o recurso mais importante no jogo, é necessária a sua verificação constantemente, pois sua tomada de decisão pode mudar partir dela. Segue abaixo o padrão identificado:

- Nome: Múltiplos Indicadores de Vida
- Origem: -
- Contexto e Problema: O marcador de vida normalmente é um elemento de extrema importância ao decorrer do jogo, pois ele influencia diretamente na situação atual de seu personagem, fazendo com que esse recurso altere a tomada de decisão do jogador. Ainda sim, existem muitos outros elementos que precisam da atenção do jogador e basear-se somente em um ponto específico da tela pode diminuir o nível de entendimento do jogador, fazendo com que ele tome decisões erradas ou até mesmo não veja um inimigo se aproximando. A necessidade gerada pela constante verificação desse recurso tão importante em meio de tantos outros requer que este esteja sempre no campo de visão do jogador.
- Solução: Usar outras alternativas além de uma barra de vida fixada a interface de ajuda ao jogador, tais como: para jogos [MOBA](#), a aparição de uma nova barra de vida contendo menos detalhes sobre a cabeça do personagem em questão, contendo por exemplo, pequenas divisões para indicar a grandeza de sua vida sem apresentar números. Já para jogos [FPS](#), onde não se tem visão do próprio personagem, a utilização de sinais luminosos ao redor da tela ou o escurecimento dos cantos da tela ao levar dano de algum inimigo e sua vida chega a certo ponto crítico.
- Princípio Heurístico: Flexibilidade e Eficiência de Uso, Visibilidade de Status do Sistema.
- Exemplos: Defense of the Ancients, League of Legends, Heroes of the Newerth, Defense of the Ancients 2, Heroes of the Storm, Magika: Wizard Wars, Smite, TOME: Immortal Arena, Strife, Arena of Fate, Guardian of Middle-Earth, Supernova, Duket Nuken: 3D, Counter-Strike: Global Offensive, Left4Dead, Far Cry 3, Bioshock, Bioshock Infinite.

A consolidação desse padrão ao gabarito se deu primeiramente a partir da necessidade gerada ao ter que manter o recurso “barra de vida” sempre no campo de visão do jogador. A partir desse problema, remeteu-se a heurística de usabilidade de [Nielsen \(1994\)](#) "flexibilidade e eficiência de uso", proporcionando ao jogador uma maior eficácia e eficiência na execução de ações. Posteriormente, a análise na lista de jogos. Ao analisá-la se nota que em muitos jogos esse recurso aparece diversas vezes em uma mesma tela, cobrindo todo o campo de visão do jogador.

Esse padrão foi descoberto a partir da análise visual da tela do jogo Defense of the Ancients 2 ([DotA2](#)), ao analisá-lo, percebe-se que em todo e qualquer momento do jogo existe uma barra de vida a cima da cabeça de seu personagem e outra barra fixada a

interface de ajuda e alerta do jogador ou HUD. A barra a cima da cabeça do personagem oferece informações limitadas, já a barra fixada a HUD, apresenta informações detalhadas sobre a vida do jogador. Abaixo uma imagem ilustrativa demonstrando a utilização das duas barras (Figura 42).

Figura 42 – Utilização de dupla barra de vida em DotA2.



Fonte: O autor.

Ao utilizar esse padrão, o jogo se torna mais intuitivo, já que a informação necessária sempre está no campo de visão do jogador, ajudando-o a tomar decisões rápidas que podem lhe custar a vida. Até mesmo uma simples divisão da barra de vida em sessões sobre a cabeça do personagem para indicar a grandeza de sua vitalidade pode se tornar um recurso muito útil, já que no calor da batalha o jogador não irá dirigir seu foco para a HUD somente para verificar seus números exatos.

Ao sofrer algum ataque ou atacar o inimigo de maneira perigosa, a atenção do jogador deve estar totalmente focada em suas próximas ações e o posicionamento de seu personagem, tornando ainda mais essencial o uso de uma barra de vida secundária que esteja dentro desse campo de visão e que não o atrapalhe em momentos cruciais na partida. Múltiplos indicadores podem parecer redundância de informação, mas neste caso ao invés de dados desnecessários, a redundância ajuda o jogador a manter esse recurso tão importante sempre a vista.

5.2.3 Padrão Sistema de Nivelamento de Partidas Online

- Nome: Sistema de Nivelamento de Partidas Online
- Outras Menções: -
- Contexto e Problema: Jogos que exigem a participação de jogadores humanos em ambos os times podem sofrer com a desmotivação de seus jogadores após múltiplas

derrotas. Em jogos FPS, ter jogadores com melhor precisão na mira, melhor trabalho em equipe e noção de jogo em uma partida pode resultar em uma partida unilateral, onde os jogadores com níveis de familiaridade e jogabilidade menores podem se sentir desmotivados, já que suas habilidades não os trouxeram benefício algum. Isso pode acontecer também em jogos MOBA, onde normalmente partidas de cinco contra cinco ocorrem e se jogadores com melhor abstração de objetivos, mecânicas de personagens e noção de compra de itens estiverem jogando em um mesmo time, o time adversário pode ter problemas e se sentirem inferiorizados.

- Solução: Existem mais de uma abordagem para a solução desse problema, uma delas sendo o ajuste de dificuldade pelo próprio jogador, muito próximo aos níveis de dificuldade apresentados para o padrão "nível de dificuldade adaptativa". Essa abordagem faria com que o jogador escolhesse antes de uma partida por exemplo, uma dificuldade como fácil, médio ou difícil. Outra opção seria utilizar alguma ferramenta de apoio para nivelamento ao longo da jornada do jogador. Levar em conta seu nível, precisão, noção de jogo, KDA (do inglês Assassinatos/Mortes/Assistências) entre outros, podem ajudar a criar um ambiente onde haja diversão e ao mesmo tempo desafio para o jogador. Ainda existe a possibilidade de utilização do sistema de "Elo⁵".
- Princípio Heurístico: Flexibilidade e Eficiência de Uso.
- Exemplos: Defense of the Ancients, League of Legends, Heroes of the Newerth, Defense of the Ancients 2, Heroes of the Storm, Magika: Wizard Wars, Smite, TOME: Immortal Arena, Strife, Arena of Fate, Guardian of Middle-Earth, Supernova, Counter-Strike: Global Offensive, Battlefield 3.

⁵ Uma fórmula usada para classificar cada jogador após cada partida, esse número gerado é usado para parear o jogador com outros de mesma classificação ou classificações parecidas.

Figura 43 – Utilização de “Sistema de Nivelamento de Partidas Online” em Counter-Strike: Global Offensive.



Fonte: O autor.

5.2.4 Padrão Múltiplos Indicadores de Status Negativo de Personagem

- Nome: Múltiplos Indicadores de Status Negativo de Personagem
- Outras Menções: -
- Contexto e Problema: Ao decorrer de uma partida, varias forças podem ser exercidas sobre um personagem jogável, podendo ou não fazer com que o jogador perca o controle sobre seu personagem. Essa perda de controle deve ser explícita e demonstrar o que fez isso acontecer, para que não haja dúvidas para o jogador. Um exemplo disso em jogos **MOBA** acontece quando um personagem é atordoado por alguma habilidade inimiga, isso faz com que o jogador perca o controle de seu personagem por poucos segundos, fazendo com que fique parado e não reaja a nenhum comando de seu jogador. Já em jogos **FPS**, isso dá-se por exemplo quando um inimigo utiliza uma granada de cegueira, o que faz com a tela do jogador fique totalmente branca por alguns segundos e um ruído agudo seja ouvido.
- Solução: Tanto para jogos **MOBA** quanto para jogos **FPS**, indicadores tanto visuais quanto sonoros podem facilitar a compreensão de *status* negativos sobre seu personagem, o que também auxilia o jogador a reagir a tal força. A adição em **MOBAs** de marcadores de situação sobre a cabeça de seu personagem pode facilitar a compreensão de informação, juntamente com sinais luminosos nos cantos da tela, ou até mesmo se, a câmera do jogador não estiver focada em seu personagem, apresentar uma marcação na tela sobre um ataque inimigo. Já em jogos **FPS**, onde não se tem visão do próprio personagem, uma linha de texto contendo a informação de status negativo pode ser adicionada em conjunto com sinais luminosos e ruídos agudos.

- Princípio Heurístico: Correspondência entre o sistema e o mundo real, Flexibilidade e eficiência de uso, Visibilidade de Status do Sistema.
- Exemplos: League of Legends, Heroes of the Newerth, Defense of the Ancients 2, Smite, Strife, Counter-Strike: Global Offensive, Left4Dead, Battlefield 3.

Figura 44 – Utilização de “Múltiplos Indicadores de Status Negativo de Personagem” em Defense of the Ancients 2.



Fonte: O autor.

5.2.5 Padrão Reutilização de Configuração Prévia

- Nome: Reutilização de Configuração Prévia
- Outras Menções: -
- Contexto e Problema: Muitos jogos utilizam vários elementos para configuração de personagem, exemplos disso são referentes as armas iniciais em jogos FPS e características de cada personagem (runas, habilidades extras e atributos de defesa e ataque) em jogos MOBA. Refazer essa configuração todo o início de uma partida se torna desgastante e desmotivador, ainda mais que jogadores normalmente têm suas preferências e a ausência disso pode afastar a pessoa e tornar o jogo menos atrativo.
- Solução: Possibilitar ao jogador antes de um partida de pré-configurar, em jogos MOBA, a compra de itens para cada campeão, não o limitando a uma aba de compra única, pré-configurar características de seu personagem, sendo elas: atributos que auxiliam o personagem a ficar mais forte ao longo da partida, sua árvore de talentos como jogador, habilidades referentes ao jogador e não o campeão, entre outros. Já em jogos FPS, dar a liberdade para o jogador pré-selecionar suas armas favoritas

ainda no menu do jogo, para que ao entrar em uma partida, seu personagem já esteja pronto para o combate. Utilizar configurações pré-programadas pode tornar o jogo mais ágil e personalizável, sempre visando a liberdade e flexibilidade do sistema de jogo.

- Princípio Heurístico: Liberdade de controle para o usuário.
- Exemplos: League of Legends, Heroes of the Newerth, Defense of the Ancients 2, Smite, TOME: Immortal Arena, Strife, Arena of Fate, Guardian of Middle-Earth, Supernova, Counter-Strike: Global Offensive, Battlefield 3

Figura 45 – Utilização de “Reutilização de Configuração Prévia” em Heroes of the Newerth.



Fonte: O autor.

6 Considerações Finais

Este trabalho teve como objetivo a criação de um catálogo de padrões de interação e usabilidade para jogos **MOBA** e **FPS**. Tendo em vista que o mercado de jogos cada vez mais atrai pessoas de todas as idades e recitando Federoff (2002) ao dizer que jogos são diferentes de *softwares* convencionais e que precisam ser analisados com uma perspectiva diferente se deu o estudo de padrões específicos para jogos, mais especificamente para esses dois gêneros escolhidos.

A criação de um catálogo de padrões para jogos se torna necessária ao notar que empresas desenvolvedoras de jogos estão sempre competindo entre si pela aprovação de seus jogadores e jogos mais intuitivos e usuais ganham um apreço melhor entre jogadores segundo Folmer (2006). Lembrando ainda que a usabilidade não torna um jogo fácil, como apontado por Juul e Norton (2009), a facilidade de compreensão dos objetivos, a completa abstração das fraquezas e forças de seu personagem não interfere na dificuldade dos obstáculos ao longo do jogo.

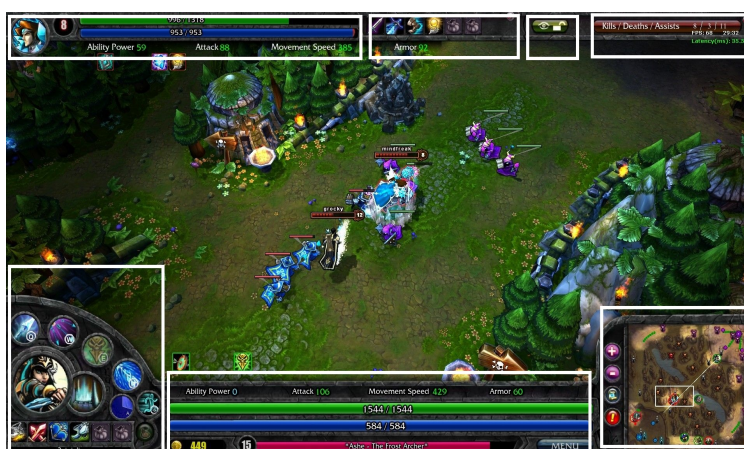
6.1 Descobertas e Contribuições

Jogos **MOBA** normalmente possuem um quantidade de mecânicas únicas maior que os demais gêneros, já que cada personagem tem suas habilidades próprias e cada habilidade contendo suas mecânicas próprias, o que os torna uma fonte rica para a identificação de padrões. Já jogos **FPS** apresentam uma padronização maior de estilo de jogo, onde mecânicas são focadas no uso de armas e não em um personagem em si. Em jogos **FPS** normalmente todos os personagens são semelhantes ou idênticos a outros dentro de um mesmo jogo. A diferença de abordagem de padrões para ambos os gêneros é visível quando observamos a necessidade de descrever uma solução diferente para cada gênero dentro do mesmo padrão ou quando notamos na escrita do padrão certa dificuldade em identificar sua implantação em algum dos gêneros.

Alguns padrões ainda podem apresentar um problema idêntico mas soluções diferentes, como é o caso do padrão “Múltiplos Indicadores de Vida”. Nesse padrão, o problema encontrasse em manter a “barra de vida” sempre em foco e para jogos **MOBA** e **FPS** utilizar-se da *HUD* para manter esse recurso sempre a vista não soluciona o problema, já que devem existir mais marcadores. A solução difere-se pelo próprio estilo de câmera, visto que para jogos **MOBA** consegue-se ver o personagem, adicionar uma “barra de vida” secundária pode solucionar o problema, já em jogos **FPS** onde não se tem visão de seu personagem é preciso utilizar-se de outros recursos, tais como áudio, marcadores vermelhos (representando sangue) em torno da tela, entre outros.

Outra descoberta foi observada ao analisar jogos antigos e novos, onde se nota uma adaptação e refatoração de elementos de interface. Um desses casos ocorre no jogo League of Legends, onde sua interface foi sendo atualizada conforme novos padrões e novas práticas foram sendo descobertas. Essa evolução é claramente visualizada nas imagens a baixo, sendo a primeira referente a fase *beta* do jogo (Figura 46) contendo uma *HUD* abrangendo espaço desnecessário na tela, a segunda referente aos três primeiros anos do jogo no mercado (Figura 47) ainda tendo informação espalhada pela tela, e a terceira referente a interface atual (Figura 48) reunindo toda a informação necessária em um pequeno número lugares.

Figura 46 – Interface referente a fase *beta* do jogo League of Legends.



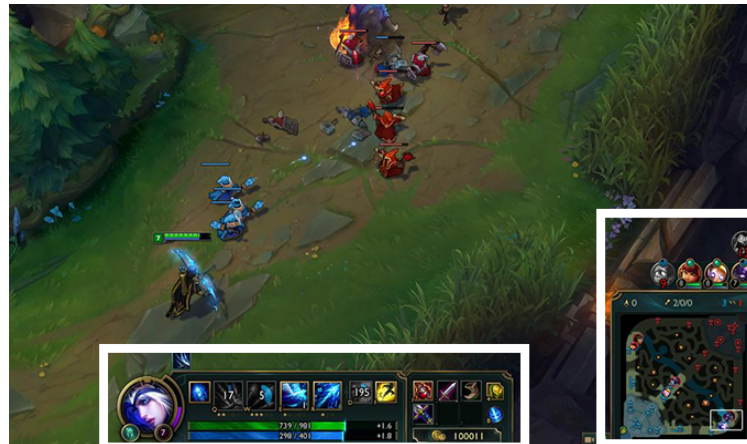
Fonte: <https://streamie.com.br/wp-content/uploads/2016/10/img-old-league-1.jpg>

Figura 47 – Interface referente aos primeiros anos do jogo League of Legends.



Fonte: https://i.ytimg.com/vi/6Qg-64D9p_k/maxresdefault.jpg

Figura 48 – Interface atual do jogo League of legends.



Fonte: O próprio autor

6.2 Limitações e Trabalhos Futuros

A análise, identificação e validação dos padrões foram realizadas exclusivamente pelo autor, limitando a confiabilidade e a abrangência dos padrões propostos. Uma solução para trabalhos futuros englobaria a aplicação de avaliações qualitativas e quantitativas juntamente com especialistas em design de jogos, inspeções de interface em busca de padrões, além de ter os textos revisados pela comunidade de jogos, tornando assim, o catálogo mais concreto e melhor aceito por empresas interessadas em aumentar a usabilidade de novos jogos novos ou atualizações em jogos já presentes no mercado.

Outra oportunidade ainda se abre na avaliação da evolução de jogos de grandes franquias, assim como jogos que recebem atualizações constantes, como no caso de jogos como League of Legends e DotA2 para jogos MOBA e Counter Strike: Global Offensive para jogos FPS.

Outro ponto volta-se para a aplicação dos padrões apresentados em diferentes gêneros de jogos. A variedade de gêneros pode tornar necessária a refatoração da solução dos padrões, fazendo com que os padrões sejam catalogados por gênero, visto que algumas vezes jogos MOBA e FPS apresentam soluções são diferentes. Já que segundo Federoff (2002), jogos não devem ser tratados como projetos de *software* comuns.

Ainda existe a pequena quantidade de diferenciação dentro dos jogos MOBA, visto que esse ainda é considerado um gênero recente. Já para jogos FPS, essa quantidade seria considerada pequena em comparação à quantidade de jogos disponíveis no mercado atual. Descartando ainda a plataforma móvel, que apresenta suas próprias limitações ligadas a diferença de jogabilidade e interação. O mercado para jogos MOBA têm crescido constantemente, trazendo consigo novos estilos de jogos que não se prendem somente a

um gênero, o que abre a possibilidade de aplicações do catálogo em jogos contendo o gênero estudado em conjunto com outros gêneros.

Referências

- ALEXANDER, C.; ISHIKAWA, S.; SILVERSTEIN, M. Pattern languages. *Center for Environmental Structure*, v. 2, 1977. Citado na página 21.
- ASSOCIATION, E. S. *Essential Facts About the Computer and Game Industry*. 2017. Disponível em: <http://www.theesa.com/wp-content/uploads/2017/09/EF2017_Design_FinalDigital.pdf>. Citado na página 39.
- BARBOSA, S. D. J.; SILVA, B. S. da. *Interação humano-computador*. [S.l.]: Elsevier, 2010. Citado na página 29.
- BARTHOLL, A. First person shooter. *Space Time Play*, Springer, p. 318–319, 2007. Citado 2 vezes nas páginas 27 e 28.
- BATTAIOLA, A. L. Jogos por computador–histórico, relevância tecnológica e mercadológica, tendências e técnicas de implementação. *Anais da XIX Jornada de Atualização em Informática, SBC*, v. 2, p. 83–122, 2000. Citado 2 vezes nas páginas 25 e 26.
- BORCHERS, J. O. A pattern approach to interaction design. *Ai & Society*, Springer, v. 15, n. 4, p. 359–376, 2001. Citado 2 vezes nas páginas 33 e 34.
- FEDEROFF, M. A. *Heuristics and usability guidelines for the creation and evaluation of fun in video games*. Tese (Doutorado) — Citeseer, 2002. Citado 4 vezes nas páginas 22, 35, 79 e 81.
- FOLMER, E. Usability patterns in games. *Future Play*, v. 6, 2006. Citado 6 vezes nas páginas 21, 36, 53, 61, 62 e 79.
- Game Accessibility Guidelines. *A straightforward reference for inclusive game design*. 2012. Disponível em: <<http://gameaccessibilityguidelines.com/>>. Citado 7 vezes nas páginas 62, 63, 64, 65, 66, 67 e 69.
- GAMMA, E. et al. *Design patterns: elements of reusable object-oriented software*. [S.l.]: Pearson Education, 1994. Citado na página 28.
- GAUDIOSI, J. Riot games’ league of legends officially becomes most played pc game in the world. *Forbes. Jul*, v. 11, p. 2011, 2012. Citado na página 27.
- HUIZINGA, J. *Homo Ludens IIs 86*. [S.l.]: Routledge, 2014. Citado na página 25.
- JUUL, J.; NORTON, M. Easy to use and incredibly difficult: on the mythical border between interface and gameplay. In: ACM. *Proceedings of the 4th international conference on foundations of digital Games*. [S.l.], 2009. p. 107–112. Citado 2 vezes nas páginas 34 e 79.
- KREIMEIER, B. *The case for game design patterns*. 2002. Citado 8 vezes nas páginas 33, 35, 53, 54, 56, 59, 63 e 64.

- MALONE, T. W. Heuristics for designing enjoyable user interfaces: Lessons from computer games. In: ACM. *Proceedings of the 1982 conference on Human factors in computing systems*. [S.l.], 1982. p. 63–68. Citado na página 34.
- NACKE, L. From playability to a hierarchical game usability model. In: ACM. *Proceedings of the 2009 Conference on Future Play on@ GDC Canada*. [S.l.], 2009. p. 11–12. Citado na página 35.
- NIELSEN, J. *Usability engineering*. [S.l.]: Elsevier, 1994. Citado 8 vezes nas páginas 21, 29, 30, 35, 54, 56, 71 e 73.
- TOXBOE, A. Ui patterns. <http://ui-patterns.com/>, Acesso em 14 de novembro de 2015. Citado na página 30.

Índice

CAPES, 49

DotA, 28, 41, 42

DotA2, 13, 42, 43

E3, 46

FPS, 21, 23, 24, 28, 29, 41, 44, 45, 57

IHC, 23, 30, 52

MMORPG, 44

MOBA, 13, 21, 23, 24, 28, 29, 41, 43, 44,
55, 57

RPG, 28, 44

RTS, 28, 44

UFPEL, 52

UNIPAMPA, 52