



**CENTRO UNIVERSITÁRIO LUTERANO DE PALMAS**

*Recredenciado pela Portaria Ministerial nº 3.607, de 17/10/05, D.O.U. nº 202, de 20/10/2005*  
ASSOCIAÇÃO EDUCACIONAL LUTERANA DO BRASIL

Djonathas Carneiro Cardoso

**CODE LIVE:** Gamificação de um ambiente virtual de programação

Palmas - TO  
2015

Djonathas Carneiro Cardoso  
**CODE LIVE:** Gamificação de um ambiente virtual de programação

Trabalho de Conclusão de Curso (TCC) elaborado e apresentado como requisito parcial para obtenção do título de bacharel em Sistemas de Informação pelo Centro Universitário Luterano de Palmas (CEULP/ULBRA).

Orientador: Prof. M.Sc. Jackson Gomes Souza.

Palmas - TO  
2015

Djonathas Carneiro Cardoso  
**CODE LIVE:** Gamificação de um ambiente virtual de programação

Trabalho de Conclusão de Curso (TCC) elaborado e apresentado como requisito parcial para obtenção do título de bacharel em Sistemas de Informação pelo Centro Universitário Luterano de Palmas (CEULP/ULBRA).

Orientador: Prof. M.Sc. Jackson Gomes Souza.

Aprovada em: xxxxxxx de XXXX.

**BANCA EXAMINADORA**

---

Prof. M.Sc. Jackson Gomes Souza  
Centro Universitário Luterano de Palmas

---

Prof. M.Sc. Fabiano Fagundes  
Centro Universitário Luterano de Palmas

---

Prof. M.Sc. Parcilene Fernandes de Brito  
Centro Universitário Luterano de Palmas

Palmas - TO  
2015

## AGRADECIMENTOS

Chega-se ao fim uma árdua jornada de 4 anos e meio, tantas histórias, tantas memórias. É até difícil nomear agradecimentos a todos que contribuíram para que eu chegasse aqui, mas tentarei fazer o meu melhor, espero que não esquecer ninguém.

Primeiramente, devo meus agradecimentos à Deus por ter me proporcionado forças e motivação para sobreviver a todos os percalços da vida e me manter firme no caminho certo. Em segundo, e não menos importante, devo agradecimentos à minha família, meus pais Alaércio e Dorvalina, que sempre acreditaram em mim e me ajudaram de forma que eu pudesse sair do interior, mudar para a capital e pagar as primeiras mensalidades da minha graduação.

Agradeço aos colegas de curso e de trabalho que me ajudaram a não me sentir sozinho e triste nos meus primeiros meses nessa nova cidade. Porém, não posso deixar de agradecer um grupo que trilhou comigo essa estrada da tecnologia da informação, os meus colegas e grandes amigos Gabriel Gusmão, Sérgio Barros e Victor Eduardo. Formávamos um “quarteto fantástico” (rsrs), sempre fazendo os trabalhos juntos, compartilhando as dificuldades, as madrugadas perdidas, mas tudo contribuiu de forma positiva para que chegássemos até aqui!

Chegando a reta final, não poderia deixar de agradecer o meu grande orientador e mentor Jackson Gomes, que me acompanha neste trabalho há quase 1 ano e meio, sempre com seu jeito culto de falar e com super ideias para os trabalhos, que geralmente eu só fazia pela metade (rsrs)...

Por fim, queria deixar meu agradecimento à um grupo de amigos e irmãos da minha igreja que contribuíram indiretamente no meu trabalho, me mostrando os caminhos certos da vida e assim me mantendo focado no desenvolvimento deste trabalho.

## RESUMO

O presente trabalho busca transformar um então ambiente virtual de programação chamado JLive 2 em um ambiente gamificado e colaborativo. Intitulado Code Live, este projeto aplica conceitos e elementos da gamificação juntamente com o uso do universo da saga *Star Wars* como metáfora, para transformar o ambiente do JLive 2 em um sistema gamificado. O Code Live é composto por desafios, que são enunciados de problemas do cotidiano onde a missão do jogador é solucioná-los utilizando seus conhecimentos básicos de programação. Ao vencer os desafios os jogadores têm como recompensa ganho de pontos de experiência e de moeda do jogo.

PALAVRAS-CHAVE: programação; jogo; gamificação; jlive.

## LISTA DE FIGURAS

Figura 1 - Elementos de jogos.....	8
Figura 2 – Osciloscópio executando <i>Tennis for Two</i> (EDMAN, 2011).....	11
Figura 3 - <i>Noughts And Crosses</i> executando no <i>EDSAC Simulator</i> (OXO, 2014) ....	11
Figura 4 – Modelo de classificação de jogos eletrônicos (LOPES, [s.d.]) .....	12
Figura 5 – À esquerda, <i>Street Fighter Alpha 2</i> (Capcom, 1998), e à direita, <i>The Legend of Zelda: Ocarina of Time</i> (Nintendo, 1998) .....	12
Figura 6 – À esquerda, <i>No One Lives Forever 2: A Spy in H.A.R.M.'s Way</i> (Monolith, 2002), e à direita, <i>Tomb Raider: Chronicles</i> (Core Design, 2000).....	13
Figura 7 – (1) <i>GTA III</i> (Rockstar North, 2002), (2) <i>Half-Life: Counter-Strike</i> (The CS Team, 2000) e (3) <i>Dark Age of Camelot</i> (Mythic Entertainment, 2001).....	14
Figura 8 – Tela de sorteio de tema (esquerda) e tela de resolução de pergunta (direita) do <i>Perguntados</i> (Etermax, 2014) .....	15
Figura 9 – Tela parcial da página inicial do portal Escola Games .....	16
Figura 10 – Tela principal do Uno & Friends (GAMELOFT, 2013) .....	18
Figura 11 – Tela inicial do jogador com seu quadro geral de progresso no Duolingo (DUOLINGO, 2014).....	19
Figura 12 – <i>Ranking</i> do <i>Duolingo</i> .....	21
Figura 13 – Tela do jogo <i>Angry Birds</i> (ROVIO ENTERTAINMENT, 2009).....	27
Figura 14 – Tela do jogo <i>Flappy Bird</i> (.GEARS STUDIOS, 2013).....	28
Figura 15 – As seis etapas para implantação de gamificação (KUUTTI, 2013) – Adaptado.....	29
Figura 16 – Funcionamento dos Laços Interligados (WERBACH; HUNTER, 2012) - Adaptado.....	32
Figura 17 – Tela do xLogo (SOARES, 2013) .....	35
Figura 18 – Tela de Code Hunt (MICROSOFT RESEARCH).....	37
Figura 19 – Tela de desafio do curso de AngularJS no Code School .....	37
Figura 20 – Tela de desafio do curso de AngularJS no Code School .....	38
Figura 21 – Personagens principais dos filmes <i>Star Wars</i> (LUCASFILM, 2008).....	42
Figura 22 – Página principal da comunidade <i>Star Wars Wiki</i> (STAR WARS WIKI, 2015d). .....	43
Figura 23 – Tela do editor de código do JLive2.....	45
Figura 24 – Conceito básico do ambiente do Code Live .....	56
Figura 25 – Ciclo de Desafios do Code Live .....	57
Figura 26 – Layout da interface do Code Live.....	70
Figura 27 – Tela Lista de Desafios.....	71
Figura 28 – Caixa de desafio.....	72
Figura 29 – Tela de detalhes do desafio .....	73
Figura 30 – Tela de compra de duelo.....	74
Figura 31 – Tela de criação de desafios .....	75
Figura 32 – <i>Modal</i> de detalhes do desafio na visão de seu autor .....	77
Figura 33 – Tela de resolução do desafio (Modo Desafio).....	78
Figura 34 – <i>Modal</i> de compilação, validação e resultados do desafio .....	79
Figura 35 – Tela de resolução de duelo (Modo Duelo) .....	80
Figura 36 – Tela de Perfil do Jogador .....	81

## LISTA DE TABELAS

Tabela 1 – Estrutura da Ordem Jedi. ....	42
Tabela 2 - Divisão de Classes de Experiência .....	63
Tabela 3 – Conquistas do Code Live .....	63
Tabela 4 – Poderes .....	64
Tabela 5 – Classificação dos níveis de dificuldade dos desafios .....	65
Tabela 6 – Requisitos e recompensas para criação de desafios .....	66
Tabela 7 – Quantidade mínima de desafios por nível de dificuldade .....	67
Tabela 8 – Requisitos e recompensas para jogar desafios.....	68
Tabela 9 – Valores limites das apostas de acordo .....	69
Tabela 10 – Relação entre cores e níveis de dificuldade .....	72

## SUMÁRIO

1	INTRODUÇÃO .....	4
2	REFERENCIAL TEÓRICO .....	7
2.1.	Jogos.....	7
2.1.1.	Elementos de Jogos .....	7
2.1.2.	Jogos Eletrônicos .....	10
2.2.	Gamificação .....	17
2.2.1.	Definições de Gamificação .....	17
2.2.2.	Elementos de Gamificação.....	17
2.2.3.	Digital Game-Based Learning (DGBL) .....	22
2.2.4.	Gamificação em Ambiente de Aprendizagem.....	26
2.2.5.	Implantando Gamificação.....	29
2.3.	Trabalhos Relacionados.....	35
2.3.1.	Linguagem de Programação Logo .....	35
2.3.2.	Code Hunt .....	36
2.3.3.	Code School .....	37
3	MATERIAIS E MÉTODOS.....	41
3.1.	Materiais.....	41
3.1.1.	AngularJS.....	41
3.1.2.	Star Wars .....	41
3.1.3.	Star Wars Wiki.....	43
3.1.4.	JLive 2 .....	44
3.2.	Métodos.....	45
3.2.1.	Definir objetivos do negócio .....	46
3.2.2.	Delimitar Comportamentos Alvo .....	47
3.2.3.	Descrever Jogadores .....	48
3.2.4.	Planejar Ciclos de Atividades .....	48
3.2.5.	Garantir Diversão .....	49
3.2.6.	Implantar Ferramentas Apropriadas.....	49
4	RESULTADOS E DISCUSSÃO.....	51
4.1.	Definir Objetivos de Negócio .....	51
4.2.	Delimitar Comportamentos Alvo .....	53
4.3.	Descrever Jogadores .....	55
4.4.	Planejar Ciclos de Atividades .....	56
4.4.1.	Ciclo de Desafios.....	56
4.4.2.	Ciclo de Evolução.....	58
4.5.	Garantir Diversão .....	59
4.6.	Implantar Ferramentas Apropriadas.....	60
4.6.1.	Elementos de Gamificação.....	60
4.6.2.	Elementos do Universo <i>Star Wars</i> .....	61
4.6.3.	Game Design.....	62
4.6.4.	Desenvolvimento do Protótipo.....	70
5	CONSIDERAÇÕES FINAIS .....	83
5.1.	Trabalhos Futuros .....	83
6	REFERÊNCIAS BIBLIOGRÁFICAS .....	85

# 1 INTRODUÇÃO

Segundo Gomes, Henriques e Mendes (2008), a programação “é muito mais do que a escrita de um conjunto de linhas de código numa dada linguagem, é uma arte e uma ciência”. Arte porque, segundo eles, existem diversas maneiras de codificar uma instrução com uso da criatividade. Também é considerada uma ciência, pois é constituída por um conjunto de regras orientadoras, sendo necessário o uso de lógica, além da existência de processos rigorosos de programação que asseguram a eficiência, economia e utilidade dos programas gerados.

Existem diversos cursos profissionalizantes, técnicos e de graduação que trabalham com o aprendizado de programação. O ensino das linguagens de programação tem como objetivo desenvolver nos indivíduos a capacidade necessária para a criação de programas capazes de resolver problemas reais.

A programação impõe um estudo bastante diferente, exigindo uma prática intensiva, compreensão dos assuntos e reflexão (GOMES; HENRIQUES; MENDES, 2008). O ato de programar comumente é feito através de IDE's (*Integrated Development Environment*, ou Ambiente Integrado de Desenvolvimento, em português) que fornecem recursos que permitem agilizar o processo de desenvolvimento de software.

Mesmo com as facilidades que as IDE's podem proporcionar à programação, ainda são necessários elementos que permitam algum tipo de aprendizado real aos usuários. Diversos estudos mostram que cada vez mais os jogos estão servindo tanto como ferramenta de auxílio na aprendizagem como também sendo o próprio meio de aprendizagem.

Até a primeira metade do século XX, a escola tinha uma grande dificuldade em entender os jogos como parte integrante do ambiente escolar (MELLO apud ARAÚJO, 2011). Os jogos eram vistos apenas como empecilho ao aprendizado e foco dos alunos, sendo então não valorizados pela escola como ferramentas educacionais.

Já na segunda metade do século XX, os jogos passam a ser um tipo de atividade reconhecida por pesquisadores da educação e pelos próprios educadores

como essencial na prática pedagógica e, também, como parte dos conteúdos curriculares (ARAÚJO, 2011). A partir disso, novos conceitos surgiram atribuindo os jogos à educação, como o conceito DGBL (*Digital Game-Based Learning*, ou Aprendizado Baseado em Jogos Digitais, em português). Esse estilo de aprendizado envolve atividades que podem variar desde completar tarefas muito simples até o desenvolvimento de habilidades de resolução de problemas complexos (COFFEY, [S.d.]).

Pensando nessa nova realidade entre jogos e educação, começou também a surgir uma grande diversidade de jogos educacionais. Jogos educacionais são jogos elaborados especificamente para ensinar determinados assuntos enquanto se joga. Segundo Tarouco et al. (2004) “os jogos podem ser ferramentas instrucionais eficientes, pois eles divertem enquanto motivam, facilitam o aprendizado e aumentam a capacidade de retenção do que foi ensinado, exercitando as funções mentais e intelectuais do jogador”.

Em 2002 surge um novo conceito dentro do contexto dos jogos, criado por Nick Pelling (MARCZEWSKI, 2013), a gamificação (termo em português), que trouxe a inclusão de mecânica, estilo, pensamento e/ou técnicas de design de jogos em ambientes de não-jogo (ZICBERMANN; CUNNINGHAM, 2011; KUUTTI, 2013). Posteriormente a gamificação passou a ser aplicada também em ambientes de aprendizagem, inclusive de programação. O uso de gamificação em ambientes de programação é algo já discutido e apresentado em alguns projetos, como o *Code Hunt* (discutido na seção 2.3.2).

Neste mesmo sentido, o presente trabalho busca trazer um maior nível lúdico a um ambiente virtual de programação chamado JLive 2. O JLive 2 é um projeto de desenvolvimento de um ambiente de programação totalmente online feito pelo acadêmico, e autor deste trabalho. Esse ambiente permite a codificação e execução de algoritmos construídos a partir de uma linguagem semelhante ao Java, porém mais simplificada e com alguns recursos em português.

Assim surge o projeto intitulado Code Live que, através da aplicação de conceitos e elementos da gamificação juntamente com o uso do universo da saga *Star*

*Wars* como metáfora, busca transformar o ambiente do JLive 2 em um sistema gamificado.

Para atingir o objetivo pretendido por este trabalho, o mesmo foi dividido em três momentos. No primeiro momento ocorreu o levantamento do referencial teórico, com foco nos conceitos que envolvem jogos e gamificação. No segundo momento foram definidos os materiais e os métodos para o projeto. Os materiais são os recursos técnicos utilizados para contribuir na definição e desenvolvimento do projeto. Já os métodos contemplam a metodologia utilizada no desenvolvimento e na implantação da gamificação no projeto. E no terceiro e último momento foi onde foram obtidos os resultados, tendo como resultado maior um protótipo funcional da aplicação.

## 2 REFERENCIAL TEÓRICO

Nessa seção serão abordadas definições, explicações e exemplificações dos conceitos-chave do projeto: Jogos (seção 2.1) e Gamificação (seção 2.2). Além disso, serão também apresentados alguns trabalhos que têm relação com este projeto (seção 2.3).

### 2.1. Jogos

Um jogo é uma forma de arte na qual os participantes, denominados jogadores, tomam decisões em busca de um objetivo (Costikyan apud Salen e Zimmerman, 2004). (HOUAISS, 2001) define um “jogo” como:

- “atividade cuja natureza ou finalidade é a diversão, o entretenimento”;
- “atividade, submetida a regras que estabelecem quem vence e quem perde”;
- “conjunto de peças, instrumentos etc. para jogar”; e
- “brincadeira”.

Existem jogos praticados individualmente como também por dois ou mais jogadores que jogam entre si como adversários ou cooperativamente com grupos de adversários.

Crawford, Peabody e Loper (2003) dividem os jogos em cinco, do que eles chamam de regiões, que foram divididas de acordo com suas características. Essas “regiões” classificam os jogos pela forma que ele é jogado, sendo elas: jogos de tabuleiro, jogos de cartas, jogos atléticos, jogos infantis e jogos eletrônicos.

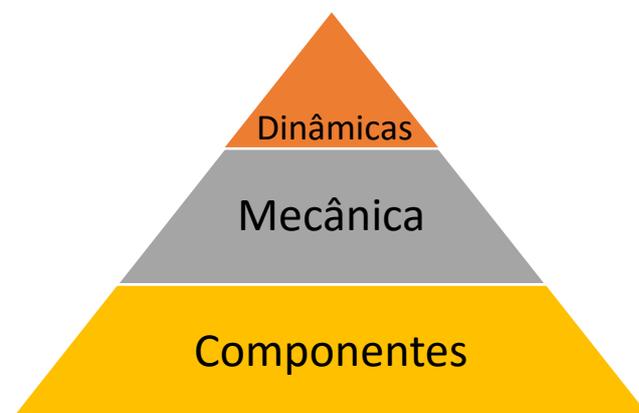
Além dos fins recreativos, em alguns casos, os jogos podem ser utilizados como parte de instrumentos educacionais. Jogos, geralmente, envolvem estimulação mental ou física e muitas vezes ambos. Uma atividade pode ser considerada um jogo quando esta possui os chamados **elementos de jogos**, que são características específicas de jogos. A seção a seguir dá mais detalhes sobre esses elementos.

#### 2.1.1. Elementos de Jogos

Werbach e Hunter (2012, p. 77) definem e dividem os elementos de jogos em três categorias: Dinâmicas, Mecânica e Componentes, que são organizados em forma de

pirâmide, sendo a Dinâmica o mais alto nível e os Componentes o mais baixo. A figura a seguir ilustra a pirâmide dos elementos de jogos.

**Figura 1 - Elementos de jogos**



**Dinâmicas** são o mais alto nível da pirâmide de elementos de jogos de Werbach e Hunter (Figura 1). Segundo Kuutti (2013) “dinâmicas são elementos que estão presentes em quase todos os jogos”. As cinco dinâmicas são (WERBACH; HUNTER, 2012):

- **Restrições:** são utilizadas para limitar a liberdade dos jogadores, a fim de criar escolhas significativas e problemas interessantes;
- **Emoções:** são um reforço emocional que mantém as pessoas jogando, podendo existir diferentes tipos de emoções, como alegria, tristeza, diversão e sentimento de realização;
- **Narrativa:** é a estrutura que torna o jogo coerente, ela não precisa ser explícita, como uma história, o importante é que o jogador sinta que sua experiência no jogo tem um propósito final;
- **Progressão:** é a ideia de dar aos jogadores a sensação de avançar e melhorar, ao invés de fazer a mesma coisa repetidas vezes, é importante ser capaz de progredir; e
- **Relacionamentos:** referem-se aos jogadores terem interação com os outros jogadores, podendo ser com os amigos, companheiros ou adversários.

A **Mecânica** representa os processos básicos que permitem ao jogador progredir no jogo. Cada mecânica é uma forma de obter uma ou mais dinâmicas. A mecânica também pode ser uma forma de obter novos jogadores ou manter os mais experientes envolvidos no jogo (WERBACH; HUNTER, 2012). Pode-se identificar dez importantes mecânicas de jogo, como definem (WERBACH; HUNTER, 2012):

- **Desafios:** são objetivos que o jogador deve cumprir;
- **Chance:** é o elemento sorte do jogo que cria uma sensação de surpresa e incerteza;
- **Cooperação e competição:** são aplicadas em conjunto para a criação de um sentimento de ganhar e perder, já que os jogadores trabalham em equipe ou um contra o outro, a fim de ganhar;
- **Feedback:** permite ao jogador visualizar seu progresso no jogo;
- **Aquisição de recursos:** é o ato de coletar coisas que ajudarão a alcançar o objetivo do jogo;
- **Recompensa:** é um benefício que o jogador pode obter a partir de uma conquista no jogo;
- **Transação:** significa compra, venda ou troca de alguma coisa com outros jogadores ou personagens no jogo.
- **Turno:** é o tempo ou oportunidade de interação de cada jogador com o jogo, porém, muitos jogos modernos trabalham em tempo real, sem turnos; e
- **Estado de vitória:** define a vitória no jogo.

Os **Componentes** são elementos que podem ser vistos e utilizados na interface do jogo. Alguns componentes podem fazer parte de uma mecânica, como, por exemplo, tanto os **emblemas** como o **conteúdo de desbloqueio** podem ser parte de **recompensas** (KUUTTI, 2013). Os quinze componentes importantes do jogo são apresentados a seguir (WERBACH; HUNTER, 2012):

- **Conquistas:** dão ao jogador recompensas, geralmente, quando ele completa uma **missão**, que é composta por um conjunto de tarefas específicas;
- **Emblemas:** é a representação visual de uma conquista;
- **Avatar:** é a representação visual do personagem do jogador;
- **Lutas contra chefes:** são desafios muito difíceis, normalmente no final de um nível, que tem de ser derrotado a fim de progredir no jogo;
- **Coleções:** são um conjunto de itens em jogos que se relacionam entre si;
- **Combate:** é o ato de lutar e derrotar oponentes no jogo;
- **Conteúdo de desbloqueio:** são conteúdos que originalmente são bloqueados nos jogos, sendo necessário o jogador fazer uma ação específica para desbloqueá-los;

- **Presentes:** permitem o jogador dar itens ou moeda virtual para outros jogadores;
- **Quadros de líderes:** listam os jogadores de acordo com suas pontuações;
- **Níveis:** são a representação de quão bom o jogador está no jogo, que é medido, geralmente, através da quantidade de **pontos** que o jogador obtém ao realizar algumas ações no jogo;
- **Gráfico social:** permite ver amigos que também estão no jogo e ser capaz de interagir com eles;
- **Equipes:** são um conjunto de jogadores trabalhando em conjunto em prol de um mesmo objetivo; e
- **Bens virtuais:** são itens que os jogadores podem coletar e utilizar.

Dentre os diversos tipos de jogos, os jogos eletrônicos são os que geralmente atribuem os elementos de jogos apresentados por Werbach e Hunter em maior número, já que não existem tantas limitações como em outros tipos de jogos, como as limitações de ambiente físico. Esse tipo de jogo será apresentado com mais detalhes na seção a seguir.

### 2.1.2. Jogos Eletrônicos

Os jogos eletrônicos ou digitais são aqueles nos quais os jogadores interagem com o jogo através de um dispositivo eletrônico, denominado como plataforma, que pode ser um computador, celular, *videogame* etc. Gonçalves e Nunes (2010) identificam como jogos eletrônicos aqueles que são executados em um ambiente virtual através de meios eletrônicos.

Segundo Batista et al. (2007), o primeiro jogo eletrônico foi datado do ano de 1958. Criado pelo físico Willy Higinbotham com o nome *Tennis Programming*, também conhecido como *Tennis for Two*, apesar de nunca ter sido patenteado. O jogo era um muito simples: consistia em rebater a bola em uma linha horizontal na parte inferior da tela de um osciloscópio. A figura a seguir mostra um exemplo de equipamento com o jogo *Tennis for Two*.

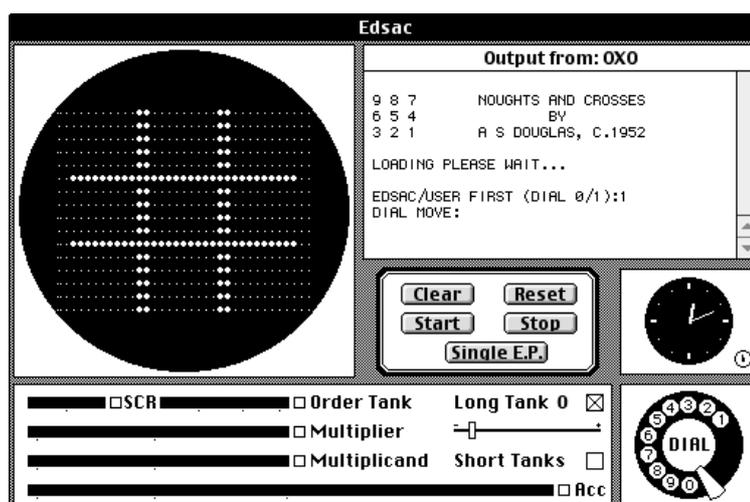
**Figura 2 – Osciloscópio executando *Tennis for Two* (EDMAN, 2011)**



Conforme pode ser observado na Figura 2, além da linha horizontal utilizada para rebater a bola, existe também uma linha vertical no centro que representa a rede. Há duas caixas com um potenciômetro e um botão para que os jogadores interajam com o jogo.

Entretanto, Winter (2013) afirma que o primeiro jogo eletrônico gráfico foi o *Noughts And Crosses*, um “jogo da velha” para o computador EDSAC, criado em 1952 por Alexander S. Douglas como uma tese de doutorado, demonstrando a interação humano-máquina para a Universidade de Cambridge. A figura a seguir mostra o jogo executando em um simulador de EDSAC.

**Figura 3 - *Noughts And Crosses* executando no *EDSAC Simulator* (OXO, 2014)**



Segundo Winter (2013), para interagir com o jogo, o jogador em questão deve escolher onde colocará seu “0” ou “X” através de um discador telefônico mecânico. Essa forma também está presente no simulador do EDSAC, conforme pode ser visto na parte inferior direita da Figura 3 o discador telefônico. Segundo Campbell-Kelly

(2006), a interface do usuário do *EDSAC Simulator* tem todos os controles e displays da máquina original EDSAC, e o sistema inclui uma biblioteca de programas originais, sub-rotinas, e software de depuração.

Segundo o engenheiro de software Gilliard Lopes, que é brasileiro e produtor do jogo FIFA 2014 pela *Electronic Arts*, um jogo eletrônico pode ser categorizado por diversos critérios, como dimensionalidade, ponto de vista, gênero e número de jogadores (LOPES, [s.d.]). A figura a seguir ilustra um modelo de organização dessa estrutura descrita por Lopes.

**Figura 4 – Modelo de classificação de jogos eletrônicos (LOPES, [s.d.])**



Conforme a Figura 4, os quatro critérios de classificação de jogos eletrônicos dividem-se em itens. Geralmente um jogo enquadra-se em um item de cada critério, com exceção do gênero. Porém nem sempre um jogo vai se encaixar em todos os critérios. Por exemplo, existem jogos que não são considerados de 1ª e nem 3ª pessoa, pelo simples fato de não conter um personagem principal que o jogador use para interagir com o jogo. A seguir serão definidos cada um dos critérios (LOPES, [s.d.]).

**Dimensionalidade** está ligada ao número de dimensões em que o jogo se expressa, desta forma classificando os jogos em 2D (duas dimensões) ou 3D (três dimensões). A figura a seguir compara os dois tipos de classificação por Dimensionalidade.

**Figura 5 – À esquerda, *Street Fighter Alpha 2* (Capcom, 1998), e à direita, *The Legend of Zelda: Ocarina of Time* (Nintendo, 1998)**



A Figura 5 mostra dois exemplos de jogos, à esquerda é exemplificado um jogo em 2D e à direita um jogo em 3D.

**Ponto de vista** considera a perspectiva através da qual a apresentação gráfica do jogo é vista pelo jogador, com relação ao personagem por ele controlado. Existem duas classes de pontos de vista: primeira e terceira pessoa. A figura a seguir exemplifica cada uma das classes.

**Figura 6 – À esquerda, *No One Lives Forever 2: A Spy in H.A.R.M.'s Way* (Monolith, 2002), e à direita, *Tomb Raider: Chronicles* (Core Design, 2000)**



Em jogos em primeira pessoa o jogador observa o jogo pelo ponto de vista do próprio personagem que ele controla, isto é, “de dentro” como mostra a Figura 6 (esquerda) onde a visão do jogador é como se estivesse “imerso” no jogo. Já em jogos em terceira pessoa acontece o inverso, o jogador observa o jogo por um ponto de vista diferente de seu personagem, isto é, “de fora”, como pode ser visto na Figura 6 (direita), onde é possível ver o personagem como se fosse um terceiro observando.

**Número de Jogadores** considera o número máximo possível de jogadores simultâneos em uma partida do jogo. A popularização da internet tornou viável a produção de jogos com suporte a múltiplos jogadores. Considera-se a existência de três modos de jogo em relação ao número de jogadores, sendo eles: *single player*,

*multiplayer* e *massive multiplayer*, conforme pode ser observado na Figura 4. A figura a seguir traz três exemplos de três jogos, cada um sendo de um dos modos descritos.

**Figura 7 – (1) *GTA III* (Rockstar North, 2002), (2) *Half-Life: Counter-Strike* (The CS Team, 2000) e (3) *Dark Age of Camelot* (Mythic Entertainment, 2001)**



No modo **single player** (Figura 7-1) apenas um jogador participa da partida, enquanto no modo **multiplayer** (Figura 7-2) até dezenas de jogadores participam da partida, de forma cooperativa ou competitiva e o **massive multiplayer** (Figura 7-3) suporta até centenas de jogadores.

**Gênero** agrupa jogos com características de jogabilidade semelhantes. A Figura 4 traz quatro exemplos de gêneros de jogos eletrônicos, mas existe uma diversidade muito maior de gêneros como, por exemplo, estratégia, RPG, esporte, simulação, quebra-cabeça, entre outros. Como os gêneros são dependentes do conteúdo para ter uma definição, eles têm mudado e evoluído conforme novos estilos de jogos eletrônicos vão sendo desenvolvidos.

A próxima seção irá descrever com mais detalhes o gênero educacional de jogos.

#### **2.1.2.1. Jogos Educacionais**

Jogos educacionais são jogos elaborados especificamente para ensinar as pessoas sobre determinado assunto, expandir conceitos, reforçar desenvolvimento e entendimento sobre um evento histórico ou cultural, ou ainda auxiliar na aprendizagem de alguma habilidade enquanto se joga.

Podem se apresentar de formas variadas, e, dependendo de seu contexto principal, podem ser utilizados até por professores nas escolas. Segundo Tarouco et al. (2004) “os jogos podem ser ferramentas instrucionais eficientes, pois eles divertem enquanto motivam, facilitam o aprendizado e aumentam a capacidade de retenção do que foi ensinado, exercitando as funções mentais e intelectuais do jogador”.

Um exemplo de jogo educacional é o *Perguntados*. Ele permite testar os conhecimentos de temas como: entretenimento, ciência, esportes, geografia, história e arte através de batalhas entre amigos, sendo assim considerado um jogo educativo e social. A figura a seguir mostra duas telas do jogo.

**Figura 8 – Tela de sorteio de tema (esquerda) e tela de resolução de pergunta (direita) do *Perguntados* (Etermax, 2014)**



No jogo *Perguntados* é possível criar partidas tão quanto receber solicitação de partida de outro jogador. Uma partida pode ter até 25 rodadas, sendo que vence o jogo quem conseguir todas as coroas primeiro ou quem tiver o número maior de coroas ao fim das rodadas. O jogador pode ganhar uma coroa ao final de 3 rodadas ou quando conseguir o item “coroa” no sorteio de temas, conforme a Figura 8 (esquerda).

Uma vez sorteado um tema, é mostrada ao jogador uma pergunta de múltipla escolha sobre o tema sorteado, conforme a Figura 8 (direita), sendo que apenas uma alternativa é a correta. Caso o jogador acerte ele ganha a chance de responder outra pergunta, assim ficando mais próximo da vitória da partida. Caso contrário, é mostrada a alternativa correta e chega a vez do oponente jogar, já que o jogo é baseado em

turnos. Também há um contador regressivo de tempo para responder, caso o jogador não responda dentro do tempo, ele perde o direito de resposta e passa a vez para seu oponente.

Também existem diversos sites voltados exclusivamente para esse gênero de jogos, porém, geralmente são mais voltados para o público infantil ou em fase de alfabetização. Tem-se como exemplo o site “Escola Games” que tem como proposta ser um site gratuito de jogos educativos para crianças a partir de 5 anos. Segundo o próprio portal, todos os jogos são desenvolvidos com acompanhamento pedagógico para que as crianças aprendam brincando. A figura a seguir mostra parte da página inicial do portal *Escola Games*.

**Figura 9 – Tela parcial da página inicial do portal Escola Games**



Conforme a Figura 9, o portal tem uma interface bastante simples e intuitiva, que mostra que seu alvo é o público infantil, principalmente se levar em consideração o uso de personagens semelhantes aos de desenhos animados. Outro ponto interessante do portal é a possibilidade de filtrar os jogos baseado na sua área de aprendizado, como português, matemática, geografia etc. Também é possível filtrar por nível de dificuldade, sendo três níveis existentes, assim o jogador pode evoluir para outro nível quando sentir que o atual já não lhe representa um desafio.

## **2.2. Gamificação**

O conceito *gamification* (no original em Inglês) foi criado por Nick Pelling em 2002 (MARCZEWSKI, 2013), apesar de que o primeiro registro documentado de uso do termo tenha sido em 2008, por Bret Terrill (FITZ-WALTER, 2013), tornando-se popular em 2010 (GOOGLE, 2014).

No Brasil, o termo foi traduzido como “gamificação”, embora há quem defenda que o termo certo em português é “ludificação” (DANVANÇO, 2013). No decorrer deste texto será utilizado o termo gamificação (sem uso de aspas duplas e itálico). Diversos autores criaram definições para gamificação, algumas das quais serão apresentadas nas próximas seções.

### **2.2.1. Definições de Gamificação**

De acordo com Fardo (2013, p. 22)

A gamificação é um fenômeno emergente, que deriva diretamente da popularização e popularidade dos games, e de suas capacidades intrínsecas de motivar a ação, resolver problemas e potencializar aprendizagens nas mais diversas áreas do conhecimento e da vida dos indivíduos.

Segundo Zicbermann e Cunningham (2011), a gamificação é a inclusão de mecânica, estilo, pensamento e/ou técnicas de design de jogos para envolver pessoas na solução de um problema. Para Kapp (2012), gamificação é “o uso de mecânicas, estéticas e pensamentos dos jogos para engajar pessoas, motivar a ação, promover a aprendizagem e resolver problemas” (tradução nossa). Já Kuutti (2013) e Deterding et al. (2011) definem que a ideia central da gamificação é utilizar elementos de jogos em um contexto não-jogo.

### **2.2.2. Elementos de Gamificação**

Segundo (KUUTTI, 2013), “apesar de todos os elementos de jogos poderem ser usados na gamificação, eles podem não ser todos igualmente relevantes. Alguns são mais adequados e úteis do que outros”. Desta forma, Kuutti, baseando-se na literatura sobre motivação e gamificação fez uma seleção dos elementos dos jogos de Werbach e Hunter que são mais relevantes a gamificação. Essa seleção, da mesma forma da pirâmide Werbach e Hunter, divide-se em três categorias: Dinâmicas, Mecânica e Componentes, que serão apresentadas e exemplificadas nas seções seguintes:

### 2.2.2.1. Dinâmicas

No nível de Dinâmicas, os cinco elementos são mantidos, conforme definidos por Werbach e Hunter. Porém isso não é uma regra, um sistema não precisa ter necessariamente todos os elementos de dinâmicas para ser considerado gamificado (KUUTTI, 2013). A seguir serão explicadas cada uma das dinâmicas.

**Restrições:** Da mesma forma que acontece nos jogos, as restrições aplicadas a gamificação tem por objetivo “criar obstáculos que os jogadores procurem resolver” (RADOFF, 2011 apud KUUTTI, 2013). As restrições podem ser apresentadas de diversas formas, como uma exigência de créditos, ou de completar um determinado objetivo. Está relacionado ao componente **conteúdo de desbloqueio**, que são conteúdos que originalmente são bloqueados, sendo necessário o jogador fazer uma ação específica para desbloqueá-los. Entretanto, Kuutti não inclui o conteúdo de desbloqueio na sua seleção de Componentes da gamificação, o que também não impede de ele ser implementado em um sistema gamificado, já que o autor deixa claro que sua seleção reúne o que é mais relevante para a gamificação, mas não é uma regra absoluta. A figura a seguir mostra um exemplo de jogo com restrições.

**Figura 10 – Tela principal do Uno & Friends (GAMELOFT, 2013)**



Conforme a tela do jogo Uno & Friends, existem dois tipos de restrições no jogo. A primeira tem relação com a necessidade de ter “fichas” no jogo para iniciar uma partida, neste caso, o jogo “cobra” duas fichas para iniciar uma nova partida (Figura 10.1). A segunda restrição é aplicada ao jogador que deseja entrar nos torneios, sendo

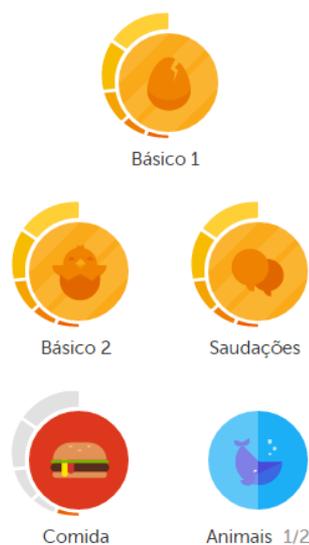
necessário que ele tenha um nível mínimo 6 (Figura 10.2) para ingressar. Para superar essas restrições, o jogador necessita vencer as partidas para que possa ganhar nível e mais fichas. A área de torneios pode ser considerada um conteúdo de desbloqueio, já que vem bloqueada originalmente no jogo e necessita que o jogador realize ações (vencer partidas e ganhar nível) para desbloqueá-la.

**Emoções:** Kuutti (2013) afirma que é mais difícil criar experiências emocionais em sistemas gamificados do que em jogos convencionais, apesar de que o autor não cita os motivos de tal dificuldade. Contudo, diferentemente dos jogos, sistemas gamificados são mais “fechados”, ou seja, mais voltado para um objetivo em específico, isso pode ser considerado um dos motivos da dificuldade em transmitir emoções nesses sistemas. No entanto, Kuutti ressalta que são experiências positivas e emoções que motivam as pessoas a jogar, por isso, deve-se tentar proporcioná-las também em sistema gamificados.

**Narrativa:** A narrativa é o que faz com que um sistema gamificado tenha coerência, para que não seja apenas um conjunto de elementos aleatórios, mas sim um sistema com propósito (KUUTTI, 2013). A gamificação não funciona se for aplicada sem pensar sobre o quadro geral e é necessário algum nível de narrativa para criar a sensação de propósito no sistema (RADOFF, 2011 apud KUUTTI, 2013).

**Progressão:** é um aspecto muito importante de sistemas gamificados, pois dá ao usuário a sensação de avançar em vez de ficar preso no mesmo lugar (KUUTTI, 2013). Ela dá ao usuário uma sensação de conquista e um objetivo de se esforçar (RADOFF, 2011 apud KUUTTI, 2013). Existem diversas formas de progressão que podem ser implementadas em sistemas gamificados, mas depende da forma que o sistema foi modelado para se definir quais tipos de progressão pode-se assumir. Por exemplo, se o sistema é dividido em fases ou capítulos, é interessante deixar isso claro ao usuário, principalmente quando ele avançar uma fase ou capítulo. Também é interessante adicionar um “quadro geral” da progressão do jogador, onde ele possa ver o que já concluiu e o que lhe falta. Um quadro geral é considerado uma forma de **feedback** sendo uma Mecânica da gamificação (KUUTTI, 2013). A figura a seguir ilustra um exemplo de quadro geral dentro de um sistema gamificado.

**Figura 11 – Tela inicial do jogador com seu quadro geral de progresso no Duolingo (DUOLINGO, 2014)**



Conforme a Figura 11, o jogo Duolingo é dividido em unidades, onde cada unidade aborda um tema e é dividida em lições. Ao completar todas as lições, a unidade é dada como concluída. Este quadro geral, em específico, tem uma importância muito maior do que mostrar o que foi já feito ou não, pois também mostra quando uma unidade precisa ser revisada. Como trata-se de um sistema para aprendizado de idiomas, os desenvolvedores avaliam como primordial, depois de um certo tempo, que o usuário pratique as unidades já concluídas como forma de reforçar o conhecimento. No quadro isso é representado pela barra de progresso vertical que fica ao lado do ícone das unidades. Quanto menos preenchida a barra, maior a necessidade de revisão. Quando a barra está cheia, o ícone da unidade fica dourado, isso significa que não há necessidade de revisão no momento.

**Relacionamentos:** é aplicado da mesma forma que nos jogos convencionais, ou seja, é aplicado para que os jogadores tenham uma interação com os outros jogadores, seja como companheiros ou adversários. Kuutti (2013) afirma que este elemento é muito popular em sistema gamificados. Jogos com multijogadores fazem parte desse conceito, já que os jogadores se relacionam como parceiros ou rivais. Alguns jogos como o GunBound da Softnyx permitem que os jogadores batalhem em partidas em que todos os jogadores são rivais ou equipe contra equipe. Também existem jogos onde o relacionamento vem na forma de *ranking* (quadro de líderes), onde geralmente todos são rivais em busca da maior pontuação. Esse tipo de relacionamento é muito utilizado em sistemas gamificados como motivador para que



**Cooperação:** Situação em que dois ou mais jogadores trabalham em conjunto em busca de objetivo em comum.

**Feedback:** É a forma do sistema gamificado apresentar a progressão do jogador.

**Recompensas:** é um benefício que o jogador pode obter a partir de uma conquista.

**Transações:** significa compra, venda ou troca de alguma coisa com outros jogadores ou personagens dentro do sistema gamificado.

### 2.2.2.3. Componentes

Da mesma forma da Mecânica, não há alterações nas definições dos Componentes dos jogos para os da gamificação, a não ser o contexto de uso. A seguir serão apresentados conceitos rápidos da seleção de Componentes dos elementos de gamificação (WERBACH; HUNTER, 2012).

**Emblemas:** segundo o dicionário Michaelis, emblema “é uma figura representativa de um conceito, sentimento ou parcialidade”. Para Werbach e Hunter (2012), emblema é a representação visual de uma conquista ou recompensa.

**Quadro de líderes:** listagem ou classificação dos jogadores de acordo com suas pontuações.

**Níveis:** representação da experiência do jogador dentro do sistema.

**Conquistas:** ato ou efeito de concluir objetivos (KUUTTI, 2013). São obtidas ao final de missões ou ao atingir algum objetivo, podendo o jogador também receber recompensas.

**Pontos:** elemento obtido pelo jogador ao realizar determinadas ações dentro do sistema gamificado, podendo influenciar nos níveis do jogador.

**Bens virtuais:** são itens que os jogadores podem coletar e utilizar.

### 2.2.3. Digital Game-Based Learning (DGBL)

O *Digital Game-Based Learning* – ou Aprendizado Baseado em Jogos Digitais (Eletrônicos) – surgiu nas últimas décadas do século 20, época em que houve grandes avanços na tecnologia ao redor do mundo (PRENSKY, 2001). Esse estilo de aprendizado envolve atividades que podem variar desde completar tarefas muito simples até o desenvolvimento de habilidades de resolução de problemas complexos (COFFEY, [S.d.]). De acordo com Deubel (2006), os gêneros dos jogos (seção 2.1.2)

devem ser levados em consideração quando os professores estão selecionando jogos para os alunos.

O primeiro relato da junção entre jogos eletrônicos e educação foi em 1973, quando Bob Jamison criou um jogo intitulado *Lemonade Stand*, que tinha um objetivo simples: fazer o máximo de dinheiro possível dentro de 30 dias com a venda de limonada (EPPER; DERRYBERRY; JACKSON, 2012). O jogo proporcionava aprendizado em administração e economia, já que tratava de todos os aspectos de um negócio, incluindo preços, controle de qualidade, controle de estoque, compras e suprimentos (MOSS, 2012).

Em 2003, o pesquisador James Gee, em seu livro “*What Video Games Have to Teach Us About Learning and Literacy?*” (em português, “O que *video games* têm a nos ensinar sobre Aprendizagem e Alfabetização”), destacou os muitos benefícios de jogos eletrônicos para a aprendizagem. Dentre esses benefícios está o de poder cometer erros sem consequências reais, assim o indivíduo é encorajado a continuar tentando. James Gee também argumenta que os jogos eletrônicos são bem adaptados para as novas formas de aprendizagem, de forma que pode-se interagir com o mundo do jogo através da exploração. Desta forma, é possível aplicar os mais diversos contextos educacionais desde que seja criado um ambiente auto didático.

Além do benefícios apresentados por James Gee, existem também seis tendências, que segundo Epper, Derryberry e Jackson (2012), podem impulsionar a adoção de aprendizado baseado em jogos, sendo elas: expectativas dos estudantes, integração de jogos e simulação, análise de dados, medalhas por aprendizagem, dispositivos móveis e o aumento da prevalência de mídia social. A seguir, cada uma destas tendências é apresentada detalhadamente (EPPER; DERRYBERRY; JACKSON, 2012).

**Expectativas dos estudantes:** Está relacionado a expectativa de aprendizado em jogos mesmo quando estes são criados apenas para fins de entretenimento. Existem jogos de perguntas e respostas que podem contribuir nos conhecimentos gerais, como o jogo “Show do Milhão” e “Perguntados”, já outros contribuem no conhecimento de mercado financeiro, como “Banco Imobiliário”. Também existem jogos compostos de histórias e enredos, alguns até baseados em momentos históricos da humanidade, como o *Age of Empires III* que se baseia na era da colonização da América, contendo fatos reais em seu enredo.

**Integração de jogos e simulação:** segundo Epper, Derryberry e Jackson (2012), simulação tem uma grande importância na aprendizagem baseada em jogos. Para que se tenha sucesso com jogos de simulação são necessários dois fatores: modelagem do sistema e representação e, fidelidade sensorial entre a simulação e o mundo real (EPPER; DERRYBERRY; JACKSON, 2012). O primeiro fator permite uma visão sobre como funciona um sistema sob diferentes condições. Já o segundo fator está relacionado ao nível de fidelidade sensorial entre a simulação e o mundo real, ou seja, o quanto o jogo se aproxima da realidade a qual ele quer retratar. Apesar que, em alguns casos, a ausência da fidelidade sensorial faz parte do apelo do jogo. Exemplo disso é o jogo *SimCity 4*, que mesmo sendo um jogo de simulação de administração de cidades, possuem elementos que fogem do mundo real, como ataques de monstros e alienígenas.

Em suma, simulações de alta fidelidade, aliadas a uma dinâmica de jogabilidade e elementos de jogo competitivos, podem aumentar o envolvimento dos alunos e melhorar a aprendizagem (EPPER; DERRYBERRY; JACKSON, 2012).

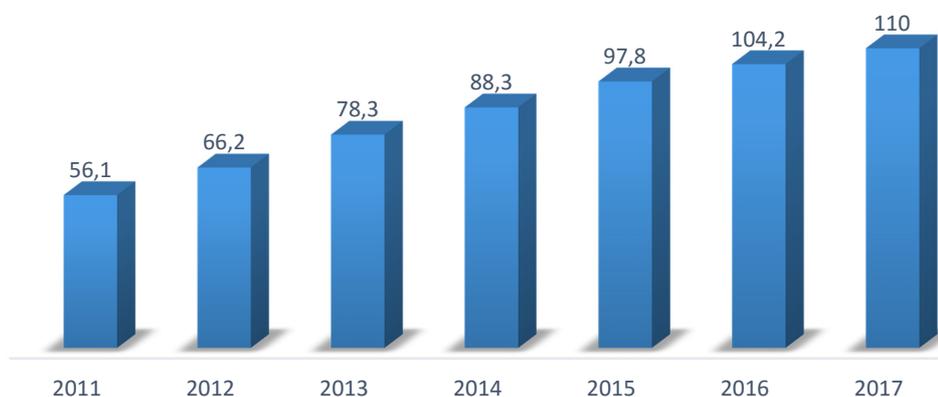
**Análise de dados:** em sua essência, os jogos são uma sucessão de interações. Como tal, jogos geram uma grande quantidade de dados e informações que podem revelar conhecimentos não só sobre o sucesso ou o fracasso do jogador, mas também sobre o trabalho em equipe, preferências de colaboração, estilos de aprendizagem, e uma variedade de outros problemas de aprendizagem. Estes dados podem ser integrados com dados de outras fontes e usados em conjunto para conduzir *insights* sobre a eficácia de uma metodologia de ensino. Instituições com a capacidade de analisar esses dados em tempo real estaria na vanguarda de uma grande oportunidade de personalizar a experiência de aprendizagem dos seus alunos a um nível sem precedentes (EPPER; DERRYBERRY; JACKSON, 2012).

**Medalhas por aprendizagem:** são fornecidas através de realizações e conquistas do estudante no jogo (EPPER; DERRYBERRY; JACKSON, 2012). Medalhas são "utilizadas para definir metas, motivar comportamentos, representam conquistas, e comunicar o sucesso em muitos contextos" (EPPER; DERRYBERRY; JACKSON, 2012). Medalhas geralmente são apenas representativas, ou seja, não influenciam no enredo do jogo. Alguns exemplos que pode-se citar de ambientes que implementaram o uso de gamificação com medalhas por aprendizagem são os portais *Duolingo* e *Code School*. Ambos são ambientes de aprendizagem, sendo o primeiro voltado para o aprendizado de idiomas e o segundo para o de programação.

**Dispositivos móveis:** Os dispositivos móveis oferecem muitos serviços que podem melhorar a experiência de aprendizado baseado em jogos. Serviços baseados em localização, como *check-ins* e *geotagging*, realidade aumentada, QR codes e acelerometria proporcionam novas formas de jogabilidade para atrair e envolver os alunos de maneira significativa (EPPER; DERRYBERRY; JACKSON, 2012). Um exemplo de jogo que explora esses serviços de geolocalização dos aparelhos móveis é o *Ingress*, que consegue misturar um ambiente virtual com o mundo real. O jogo funciona utilizando uma API do *Google Maps* onde são inseridos pontos de energia, portais e outros elementos do jogo no mapa, assim o jogador deve se mover fisicamente até o local para interagir com o jogo.

**Aumento da prevalência de mídia social:** De acordo com o relatório intitulado “*Brazil Social Media: The Mobile Middle Class Goes Social*” (em português, “A mídia social no Brasil: a classe média móvel torna-se social”), o número de usuários de redes sociais no Brasil tem crescido desde 2011 e deve continuar crescendo até pelo menos 2017 (GUIMARÃES *et al.*, 2013), conforme ilustra o gráfico a seguir.

**Gráfico 1 – Usuários de Redes Sociais no Brasil (em milhões) (GUIMARÃES *et al.*, 2013) - Adaptado**



Analisando o Gráfico 1 é possível perceber que há um crescimento médio de 9 milhões de usuários ao ano, apesar de que também há uma diminuição no ritmo de crescimento na projeção até 2017, o que representa uma taxa de cerca de 15% de redução de crescimento quando comparado com 2011. A partir desses dados, pode-se compreender as redes sociais como um ótimo ambiente para se trabalhar com aprendizagem, já que elas também são uma plataforma para os chamados jogos sociais.

Embora as tendências anteriormente descritas estejam impulsionando a adoção de aprendizado baseado em jogos, existem diversos fatores significativos que continuam a inibir a rápida adoção dessa modalidade de aprendizado, sendo eles (EPPER; DERRYBERRY; JACKSON, 2012):

- Combinar um *game design* envolvente com uma narrativa com objetivos de aprendizagem é um desafio;
- É igualmente difícil projetar uma experiência de jogo que seja acessível para todos os alunos, independentemente da sua capacidade física.
- O desenvolvimento de jogos exige múltiplas competências, não apenas o conteúdo e experiência docente.
- Não há um canal de publicação e distribuição de jogos educativos. Para os instrutores que não estão interessados ou capazes de fazer o seu próprio jogo, o tempo necessário para encontrar jogos apropriados para seus cursos e descobrir como incorporá-las em seus programas de estudo e experiências de aprendizagem podem ser fatores prejudiciais.
- Alguns professores mantêm barreiras culturais que os inibem que tentarem realizar inovações de metodologia de ensino, já que na sua visão, manter o que “funciona” é o que importa. Essa preocupação também se reflete em muitos pais de alunos, embora com menos incidência.
- O custo de desenvolvimento de um curso totalmente baseado em jogo pode ser um fator negativo para algumas faculdades e universidades.

#### **2.2.4. Gamificação em Ambiente de Aprendizagem**

Como apresentado nas seções anteriores, existem diversas tendências que mostram o quanto a gamificação pode impulsionar o aprendizado. Esta seção terá como objetivo mostrar o quanto a gamificação pode gerar resultados positivos dentro de um ambiente de aprendizagem.

Segundo Fardo (2013), “sem um conhecimento aprofundado sobre a gamificação, aplicá-la na educação pode impactar de forma não esperada os processos de ensino e aprendizagem”. É necessário entender a melhor forma de aplicar a gamificação em um ambiente de aprendizagem para que a mesma possa trazer benefícios efetivos. Com base nessa premissa, Lee e Hammer (2011) definem três áreas que a gamificação pode intervir dentro desses ambientes, sendo elas: cognitiva, emocional e social. Essas áreas serão apresentadas nas seções seguintes.

### 2.2.4.1. Área Cognitiva

Segundo o dicionário Michaelis, “cognição” é dito como ato de adquirir um conhecimento. Desta forma, os jogos podem ter essa característica através de sistemas complexos de regras para o jogador experimentar e descobrir como resolver (LEE; HAMMER, 2011). Por exemplo, no jogo *Angry Birds* o objetivo é destruir todos os inimigos do cenário derrubando torres com o lançamento de aves usando um estilingue. Para isso, os jogadores precisam “experimentar” o jogo para descobrir as propriedades físicas dos diferentes materiais das torres, a balística do estilingue, e as fraquezas estruturais de cada torre. A figura a seguir ilustra uma tela do jogo.

**Figura 13 – Tela do jogo Angry Birds (ROVIO ENTERTAINMENT, 2009)**



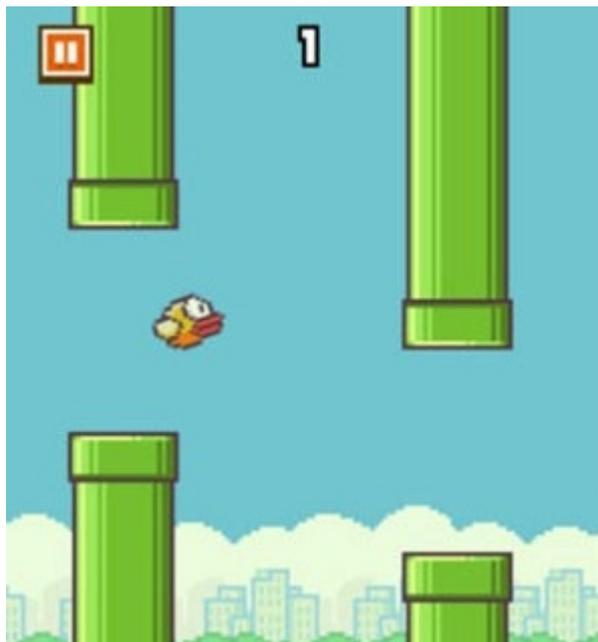
Conforme mostra a Figura 13, o jogador utiliza o estilingue para lançar o pássaro que deve colidir com a torre causando assim seu desmoronamento. O objetivo do jogo é destruir os inimigos (porcos verdes) através do desmoronamento das torres. Após a colisão, o jogador deve observar os resultados, como os danos feitos nas torres e quanto inimigos pôde destruir, para então planejar seus próximos passos e executar sua próxima jogada. Em suma, o desejo dos jogadores para vencer cada nível, através da observação dos resultados, das tentativas e erros, os torna físicos experimentais em pequena escala (LEE; HAMMER, 2011).

### 2.2.4.2. Área Emocional

Jogos podem gerar diversas emoções, como curiosidade, competitividade, frustração, alegria etc. (WERBACH; HUNTER, 2012). As emoções também ajudam os jogadores a persistirem através de experiências emocionais negativas e até mesmo transformá-las em positivas (LEE; HAMMER, 2011)

Segundo Lee e Hammer (2011), o exemplo mais dramático de emoção em um jogo é a sentida no fracasso, ainda mais em jogos com nível de dificuldade elevada resultando repetidas falhas. Um exemplo disso era o jogo *Flappy Bird*, para celulares e *tablets*, considerado um jogo altamente difícil segundo reportagem publicada no G1 (PETRÓ, 2014). A figura a seguir mostra uma tela do jogo.

**Figura 14 – Tela do jogo Flappy Bird (.GEARS STUDIOS, 2013)**



Como pode ser observado na Figura 14, o jogo não conta com gráficos robustos (em termos de nível de detalhamento e qualidade visual), muito pelo contrário, os gráficos são bastante simples, assim como o objetivo do jogo, que é controlar um pássaro (tocando na tela) sem deixá-lo colidir com os canos. Se o pássaro tocar em algum obstáculo ou cair o jogo termina. Neste momento surge a dificuldade do jogo, já que a jogabilidade (uso do toque) junto com a velocidade do pássaro, que aumenta de acordo com o tempo, e curta distância entre os canos torna o grau de sucesso do jogador quase impossível.

O sentimento causado pelo fracasso dentro de um ambiente de aprendizagem é mais negativo, pois, diferentemente do jogo, não há muitas oportunidades para o aluno “experimentar” (LEE; HAMMER, 2011). Neste caso, a gamificação deve ser aplicada para transformar o fracasso em uma parte necessária da aprendizagem e criar um ambiente no qual o esforço é recompensado (LEE; HAMMER, 2011). Isso é feito através de feedbacks rápidos, dando a possibilidade de o aluno avaliar suas próprias capacidades. “Os alunos, por sua vez, podem aprender a ver o fracasso como

uma oportunidade, em vez de tornar-se impotente, com medo ou oprimido” (LEE; HAMMER, 2011).

#### **2.2.4.3. Área Social**

Jogos podem permitir aos jogadores experimentar novas identidades e papéis, podendo também adotar papéis que explorem novos lados de si mesmos no espaço seguro de jogo (LEE; HAMMER, 2011). Por exemplo, um adolescente tímido pode se tornar um líder de um clã de um jogo RPG (*Role Playing Game*), comandando dezenas de outros jogadores em batalhas contra legiões de inimigos, demonstrando assim maior socialização do que ele consegue na vida real.

Segundo Lee e Hammer (2011), a gamificação permite que os alunos criem uma identidade diferente, dando a eles o poder de se tornarem mais sociais, além de obterem reconhecimento de suas realizações acadêmicas, que de outra forma permaneceria invisível à outros estudantes. O reconhecimento pode ser fornecido pelo professor, mas também pode-se permitir que os alunos premiem uns aos outros com a moeda do jogo. Tal projeto incentiva os alunos a reforçar o desenvolvimento de uma identidade escolar com outros estudantes, bem como consigo mesmo (LEE; HAMMER, 2011).

#### **2.2.5. Implantando Gamificação**

Agora que já foram mostrados os conceitos de gamificação, seus elementos e sua aplicação na aprendizagem, será descrito nesta seção como implantar a gamificação efetivamente em um sistema. Werbach e Hunter (2012) definiram seis etapas para a obtenção de uma gamificação eficiente. Essas etapas foram classificadas por Kuutti (2013) como um *design framework* para gamificação, de forma que ele também o nomeou como *framework D6*, pois cada uma das seis etapas iniciam com a letra “D”. A figura a seguir ilustra as seis etapas conforme definidas por Werbach e Hunter (2012).

**Figura 15 – As seis etapas para implantação de gamificação (KUUTTI, 2013) – Adaptado**



Para entender melhor o funcionamento deste *framework*, serão explicadas cada uma das etapas nas seções seguintes, para tanto, também serão feitas algumas adaptações nas traduções das etapas para melhor entendimento.

#### 2.2.5.1. Definir seus Objetivos de Negócio

Para uma gamificação eficaz, é fundamental ter um conhecimento sobre todo o sistema, seus objetivos, público-alvo, dentre outros processos, que são considerados essenciais para iniciar o desenvolvimento de qualquer sistema (WERBACH; HUNTER, 2012). Para Kuutti (2013), uma gamificação efetiva precisa de uma boa compreensão de seus objetivos que deverão ser usados como um mapa para o desenvolvimento da aplicação.

Por que gamificar o negócio? Quais as necessidades do negócio? São exemplos de perguntas que podem ser usadas na busca pela definição dos objetivos de negócio dentro da gamificação.

#### 2.2.5.2. Delimitar Comportamentos Alvo

Depois de identificar os objetivos e os motivos para a gamificação do sistema, tem-se a etapa para a delimitação dos comportamentos alvos do sistema. Essa etapa consiste em definir o que os jogadores podem fazer no sistema e como medir isso. A descrição do comportamento deve ser o mais específico, para que não haja interpretações equivocadas quando for realizada a implementação (WERBACH; HUNTER, 2012).

Alguns exemplos de comportamentos que podem ser delimitados:

- Compartilhar informações sobre o serviço no *Twitter*
- Comentar ou votar em sugestões feitas por outros
- Publicar um comentário em um fórum de discussão

Como pode ser percebido, os comportamentos se assemelham com o levantamento de requisitos de um software, já que ambos buscam delimitar as fronteiras do sistema. Um sistema com um maior número de comportamentos, também dispõe de uma maior variedade de opções e atividades disponíveis aos usuários, porém deve-se ter cuidado para não tornar o sistema complexo e confuso (WERBACH; HUNTER, 2012). Resumindo, um sistema com uma grande diversidade de ações pode ter uma boa visibilidade aos usuários, porém se estas ações estiverem organizadas da maneira adequada, poderá causar um efeito oposto.

Com posse dos comportamentos, faz-se necessário definir métricas para cada um deles. Uma métrica consiste em traduzir uma atividade em números e gerar feedback a partir disso, sendo que os números podem ser ou não transparentes ao jogador (WERBACH; HUNTER, 2012). Um exemplo de métrica transparente ao jogador seria quando ele cumpre um determinado objetivo e recebe pontos e/ou experiência por isso.

#### **2.2.5.3. Descrever seus Jogadores**

Os jogadores são os usuários do sistema gamificado. Nem todos os jogadores são iguais, sendo importante segmentá-los para que o sistema se adeque a mais de um subgrupo de jogadores (WERBACH; HUNTER, 2012). Identificar e descrever os segmentos de jogadores contribui para o rastreamento de informações como: quais tipos tem as mesmas preferências no sistema ou quais estão mais presentes. Ao saber qual tipo de jogador vai estar usando o sistema, é possível criar algo mais específico para eles (WERBACH; HUNTER, 2012).

Alguns jogos oferecem opções de segmento para o jogador, como o jogo *Dungeon Siege III*, que oferece quatro tipos de personagens, cada um com suas características, formas de jogar e histórias. Dando esse leque de opções, o jogador pode encontrar o personagem que mais se adeque a seu perfil. Da mesma forma, as plataformas gamificadas podem recorrer a segmentação de jogadores, para oferecer maior diversidade aos mesmos.

#### **2.2.5.4. Planejar Ciclos de Atividades**

Jogos sempre tem um começo e, por vezes, ter um fim, mas ao longo do caminho ocorrem uma série de *loops* e árvores de ramificação (WERBACH; HUNTER, 2012). Em outras palavras, o jogo não é simplesmente linear: Passo 1 → Passo 2 → Passo 3 → Conclusão.

A maneira mais eficiente de modelar as ações em um sistema gamificado é por ciclos de atividade, um conceito que ganhou força por ser muito empregado em redes sociais (WERBACH; HUNTER, 2012). O conceito é que uma ação provoca uma outra ação, que por sua vez provoca outra ação (KUUTTI, 2013). Como exemplo, imagine um usuário marcar um amigo em uma foto no Facebook, esta ação desencadeia uma notificação para o usuário marcado, que por sua vez posta um comentário sobre a foto, gerando então uma nova notificação ao primeiro usuário. Este exemplo é uma clara demonstração de um ciclo dentro de um sistema, já que as ações não ficam isoladas, podendo gerar diversas outras e até mesmo retornar a ação original.

Algo importante para os ciclos é dar feedback imediato para o usuário, isso motiva o usuário a executar outra ação (WERBACH; HUNTER, 2012). O objetivo é que sistema mostre aos usuários sempre que eles façam algo de bom e um feedback imediato pode proporcionar isso.

Segundo Werbach e Hunter (2012), existem dois tipos de ciclos de desenvolvimento: Laços Interligados e Escadas de Progressão. Laços Interligados descrevem, em um nível micro, o que seus jogadores fazem, por que eles fazem isso, e que o sistema faz em resposta. As Escadas de Progressão dão uma perspectiva macro na jornada do jogador. A figura a seguir ilustra a dinâmica dos Laços Interligados.

**Figura 16 – Funcionamento dos Laços Interligados (WERBACH; HUNTER, 2012)**  
**- Adaptado.**



Nos Laços Interligados, um jogador executa ações de acordo com que ele se sente motivado a fazê-las. E essas ações, por sua vez, produzem um feedback na forma de respostas do sistema, como a atribuição de pontos. Esse feedback motiva o usuário a tomar novas ações, e assim por diante. O elemento-chave aqui é feedback.

O feedback é parte do que faz jogos tão eficazes como motivadores (WERBACH; HUNTER, 2012).

Escadas de progressão refletem o fato de que a experiência do jogo muda de acordo como os jogadores se movem através dele (WERBACH; HUNTER, 2012). Isso normalmente significa um nível crescente de desafios. Na maioria dos jogos que envolvem níveis, como *World of Warcraft*, avançar do nível 1 para o nível 2 leva muito menos tempo e pontos de experiência do que mudar de nível 20 para nível 21, e por sua vez do Nível 84 para o Nível 85. Isso é feito propositalmente para que o jogo se torne cada vez mais desafiante de acordo com que a experiência do usuário aumenta.

#### **2.2.5.5. Garantir a Diversão**

“A última coisa a fazer antes de iniciar a implementação de um sistema gamificado é dar um passo para trás e fazer uma pergunta simples: É divertido?” (WERBACH; HUNTER, 2012). Durante o planejamento de um sistema gamificado, em meio de tantos processos, como definição dos elementos de gamificação, tipos de jogadores, regras, é comum que o aspecto diversão seja esquecido. Porém, “diversão é o núcleo dos jogos e gamificação” (KUUTTI, 2013), logo a diversão é um item imprescindível de ser implementado tanto em um jogo como um sistema gamificado (WERBACH; HUNTER, 2012). Quando há diversão no sistema, os jogadores serão mais propensos a retornarem e jogarem novamente, já que a diversão causa também um efeito motivador.

Os jogadores participam do sistema voluntariamente? Se não houvesse quaisquer recompensas, eles ainda estariam propensos a jogar? Werbach e Hunter ao definirem essas duas perguntas, afirmaram que elas podem contribuir na identificação do quanto o sistema é divertido, sendo que se uma das respostas for negativa, significaria que o sistema necessita de mais diversão.

#### **2.2.5.6. Implantar as Ferramentas Apropriadas**

A fase de implantação das ferramentas apropriadas é também a última fase do *framework* e também a fase que de fato vai ocorrer as implementações da gamificação. Inicialmente deve-se voltar a atenção na escolha dos elementos de gamificação que serão usados, podendo utilizar como base a seleção feita por Kuutti (Seção 2.2.2) ou mesmo selecionar a partir dos elementos de jogos (Seção 2.1.1) definidos por Werbach e Hunter. Os elementos devem ser escolhidos levando em

consideração quais melhor se adaptam as metas, comportamento alvo, tipos de jogadores, laços de atividades e diversão do sistema (KUUTTI, 2013).

Por fim, para que de fato tenha-se uma gamificação eficaz, será necessária uma equipe com uma variedade de habilidades. O que não impede de um único indivíduo faça isso, desde que ele tenha experiência em pelo menos duas das habilidades que serão apresentadas a seguir (WERBACH; HUNTER, 2012):

- **Habilidade de entender os objetivos de negócio do projeto**

Mesmo os melhores *designers* de jogos se não conseguirem entender os objetivos de negócio do sistema haverá sempre o risco de produzirem algo inútil, ou seja, que não condiz com o esperado do sistema.

- **Habilidade de compreensão do seu grupo-alvo de jogadores e as noções básicas de psicologia**

Compreender os tipos e grupos de jogadores tem grande relevância para a adesão do jogo por vários tipos de jogadores (Seção 2.2.5.3). Com posse dessas informações de tipos de jogadores aliado com um profissional com conhecimento mínimo de psicologia pode tornar o processo de identificação de perfil de jogadores mais efetivo, já que poderão ser analisadas suas características emocionais.

- **Habilidade de criação de jogos**

Ter um designer de jogos ou alguém com habilidade de criação de jogos é primordial já que um sistema gamificado tem características semelhantes a um jogo.

- **Habilidade em análise de dados**

Um especialista em análise de dados pode fazer com que os dados gerados no sistema gamificado façam sentido e se tornem úteis para serem aplicados quando necessários.

- **Habilidade de implementar a sua visão**

De nada adiantará ter alguém com a capacidade de ter uma visão se este não é capaz de a pôr em prática. Sem essa habilidade, não faz mais sentido ter as demais, uma vez que sem a efetiva aplicação da visão, as chances de sucesso são baixas.

## 2.3. Trabalhos Relacionados

Esta seção tem por finalidade trazer alguns trabalhos e projetos que se relacionam com o objetivo deste trabalho e que serviram como base ou motivação para o desenvolvimento deste. As subseções seguintes irão exemplificar melhor os trabalhos relacionados.

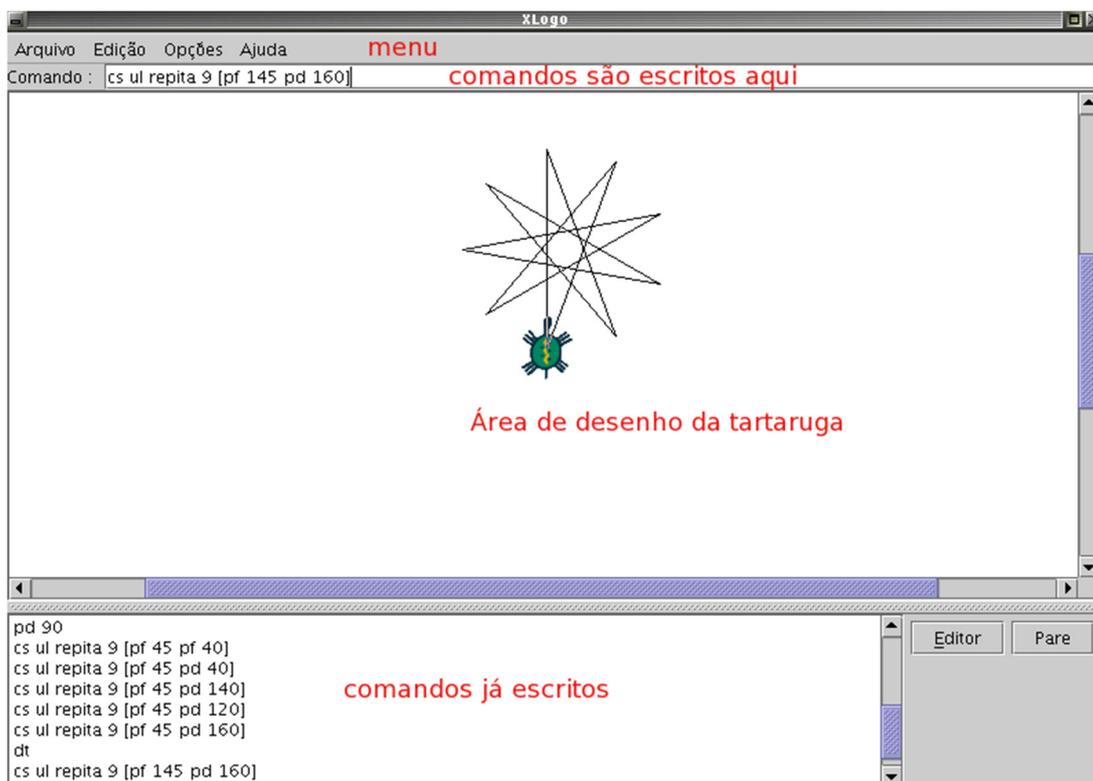
### 2.3.1. Linguagem de Programação Logo

Desenvolvida por Seymour Papert, um educador matemático, nos anos sessenta, no *MIT - Massachusetts Institute of Technology*, de Cambridge, Estados Unidos, e adaptada para o português em 1982, na Unicamp, pelo Núcleo de Informática Aplicada à Educação (NIED), a linguagem *logo* vem sendo utilizada para trabalhar com crianças e adolescentes (SOARES, 2009).

É uma linguagem considerada, ao mesmo tempo, simples e sofisticada. Do ponto de vista educacional, é uma linguagem simples, porque possui características que torna acessível o seu uso por sujeitos de diversas áreas e de diferentes níveis de escolaridade. Computacionalmente, Logo é considerada uma linguagem bastante sofisticada, por possuir características pertencentes a três paradigmas computacionais distintos: procedural, orientado a objetos e funcional. Entretanto, Logo é mais conhecido pelo paradigma procedural, especialmente, o Logo Gráfico (PRADO, 2000).

Logo Gráfico caracteriza-se pela presença de um cursor, representado pela figura de uma Tartaruga, que pode ser deslocada no espaço da tela por meio de alguns comandos básicos (*parafrente, paratrás, paradireita, paraesquerda*). A figura a seguir ilustra um dos editores de Logo executando os comandos digitados.

**Figura 17 – Tela do xLogo (SOARES, 2013)**



Essa interação gráfica entre o código e a execução proporciona que o sujeito explore diversos conceitos, tais como: noções espaciais, sequência, números em relação a distância e ao giro, estimativa, reversibilidade, operações aritméticas (adição e subtração) e geometria. Além disso, também atende determinados objetivos educacionais. Um dos principais exemplos disso é quando o sujeito necessita de um apoio figurativo para compreender e colocar em ação sua linha de pensamento em relação a resolução do problema (PRADO, 2000).

Levando em consideração a interação gráfica da Logo e seus benefícios ao aprendizado, pode-se dizer que a linguagem funciona de forma lúdica. Neste ponto é que há relação da Logo com o presente trabalho, já que ambos prezam a característica lúdica como importante. Outro ponto de relação entre os dois trabalhos é o uso de linguagens de programação, especialmente voltada para o público iniciante e para incentivo de sua prática e aprendizado, apesar de não ser uma restrição de uso para outros públicos ou finalidades.

### 2.3.2. Code Hunt

Code Hunt é um jogo para a web que ajuda no aprendizado de programação das linguagens Java e C# de uma maneira simples e divertida. O jogo consiste em diversos desafios onde são disponibilizados trechos de códigos aos jogadores, porém com algum erro de semântica. Desta forma, o jogador, a partir de resultados

esperados que o jogo fornece, deve corrigir o código da melhor forma possível. A pontuação é calculada com base na elegância, isto é, na simplicidade da solução encontrada (CANALTECH, 2014). A figura a seguir mostra a tela em que acontece a interação dentro do jogo.

**Figura 18 – Tela de Code Hunt (MICROSOFT RESEARCH)**



Conforme a Figura 18, do lado esquerdo fica localizado o trecho de código que deve ser corrigido pelo jogador e ao lado direito são apresentados os valores que serão passados na variável “x” e o resultado esperado em cada uma das iterações. Este é um jogo onde o aprendizado é baseado na tentativa e erro, onde a cada mudança no código que gere erro, o jogo mostra resultados diferentes a fim de indicar o rumo da resolução ao jogador.

### 2.3.3. Code School

Code School é uma plataforma de aprendizagem on-line que oferece diversos cursos, com foco principal em linguagens de programação. Os cursos variam do iniciante ao avançado, onde há o ganho prêmios e emblemas de acordo com o aprendizado.

Os cursos utilizam vídeos e exercícios interativos para guiar o indivíduo no seu aprendizado. Cada curso tem, pelo menos, cinco níveis. Cada nível começa com um vídeo com duração entre 10 a 15 minutos, seguido por uma série de desafios práticos que deve-se resolver para alcançar o próximo nível. A figura a seguir mostra um dos desafios dentro do curso de AngularJS do *Code School*.

**Figura 19 – Tela de desafio do curso de AngularJS no Code School**

The screenshot shows a challenge interface with the following elements:

- 1**: Challenge title "1.2 Creating a Store Module" with a "250 PTS" badge.
- 2**: Introductory text: "The Flatlanders need a store to sell their gems and more! They need it really quick, Angular will do the trick!"
- 3**: A button labeled "BUY ANSWER (100 POINTS)".
- 4**: "Objectives" section with three progress indicators, the first of which is active.
- 5**: Objective text: "Create a Module named `gemStore` so we can get started on this marketing journey."
- 6**: A button labeled "SOLUTION (-10 PTS)" and a "WATCH VIDEO" button.

A tela de desafio dos cursos no *Code School* é dividida em duas partes, na primeira (Figura 19) é apresentado o enunciado do desafio e seus objetivos, e na segunda (Figura 20) fica a área de edição do código do desafio. Cada desafio, quando concluído, gera como recompensa uma determinada quantidade de pontos (Figura 19.1). Um desafio é composto por um enunciado (Figura 19.2) e objetivos (Figura 19.5). O usuário vence o desafio ao concluir todos os objetivos (Figura 19.4) ou ele pode comprar a resposta (Figura 19.3). Se o usuário estiver com dificuldade em um objetivo em específico, também pode comprar a solução para este (Figura 19.6). A próxima figura, apresentada a seguir, representa a segunda parte da tela de desafio do *Code School*.

**Figura 20 – Tela de desafio do curso de AngularJS no Code School**

```

1 <!DOCTYPE html>
2 <html>
3 <head>
4   <link rel="stylesheet" type="text/css" href="bootstrap.min.
5   <script type="text/javascript" src="angular.min.js"></scrip
6   <script type="text/javascript" src="app.js"></script>
7 </head>
8 <body>
9   <h1></h1>
10 </body>
11 </html>
12 |

```

Slides Preview

CHECK MY WORK

Na segunda parte da tela de desafio do *Code School* (Figura 20) fica localizado o editor de código que permite ao usuário confeccionar o código para alcançar os objetivos do desafio. Alguns desafios exigem edição de múltiplos arquivos de código, por isso na parte superior da tela ficam disponíveis abas com os arquivos que podem ser editados no desafio. De acordo com o que o usuário vai editando no código, o sistema já verifica se os objetivos estão sendo cumpridos e atribui um “visto” nos concluídos. Quando o usuário concluir todos os objetivos, ele pode acionar o botão “*check my work*” na parte inferior direita da tela para então o sistema verificar se está tudo correto, atribuir a pontuação e permitir que o usuário prossiga com o curso. Caso o usuário tente usar o botão sem concluir os objetivos, o sistema irá disparar mensagens de erros nos objetivos não concluídos, com dicas para que o usuário possa concluí-los.

*Code School* é um dos projetos que mais se assemelha com o objetivo deste presente trabalho já que é um ambiente de aprendizagem de programação gamificado, também induzindo a uma prática de programação de forma lúdica. O principal diferencial entre os trabalhos seria na implantação de mais elementos de gamificação em relação ao *Code School*, como, por exemplo: quadro de líderes, níveis do jogador etc. Outro ponto que difere os sistemas é que com a implantação do quadro de líderes, o sistema proposto torna-se mais social do que o *Code School*, que não oferece esse recurso.

Baseado nos conceitos e trabalhos apresentados nas seções anteriores, o projeto proposto neste trabalho seguirá uma metodologia específica para seu

desenvolvimento. Essa metodologia abrange todas as etapas do desenvolvimento do trabalho, desde a definição, concluída nessa parte do trabalho juntamente com a primeira parte do desenvolvimento, até a etapa de testes. A seção seguinte apresenta com mais detalhes essa metodologia proposta.

## 3 MATERIAIS E MÉTODOS

O presente trabalho trata-se de uma pesquisa aplicada voltada ao contexto de Sistemas de Informação. Este capítulo apresenta os instrumentos de apoio (materiais) e os procedimentos (métodos) utilizados para o desenvolvimento deste trabalho.

### 3.1. Materiais

Os materiais utilizados neste trabalho envolveram, principalmente: livros, teses, dissertações, monografias e artigos científicos. Além disso, para o desenvolvimento do trabalho, dois materiais foram imprescindíveis: o *framework* AngularJS e o universo *Star Wars*, sendo apresentados nas suas respectivas subseções a seguir.

#### 3.1.1. AngularJS

O *AngularJS* é um *framework JavaScript open-source*, mantido pelo *Google* e por uma comunidade de desenvolvedores individuais e empresas, para resolver muitos dos desafios encontrados no desenvolvimento de *Single-Page Applications* (SPAs). As SPAs são aplicações completas, desenvolvidas em *JavaScript*, que funcionam semelhantemente a aplicações desktop.

#### 3.1.2. Star Wars

*Star Wars* (Guerra nas Estrelas, em português) é o título de uma série de seis filmes de ficção científica escritos por George Lucas. Como subprodutos surgiram também uma franquia literária, uma série de jogos eletrônicos e desenhos animados (incluindo a Nova Trilogia, sequências e adaptações literárias) baseados nas ideias do cineasta e roteirista George Lucas (STAR WARS WIKI, 2015c).

Um dos elementos de destaque em *Star Wars* é a "Força", uma energia onipresente que pode ser utilizada por aqueles com habilidade para tal (membros da ordem Jedi ou Sith) (STAR WARS WIKI, 2015a). A Força permite aos seus usuários realizar diversos feitos sobrenaturais, podendo também amplificar certas

características físicas, como velocidade e reflexos e impulso. Essas habilidades variam entre os personagens e podem ser melhoradas através de treino.

A Ordem Jedi é uma organização de guardiões da paz, composta por diversos guerreiros Jedi que são classificados por uma ordem hierárquica, as classificações são listadas na tabela a seguir (STAR WARS WIKI, 2015b).

**Tabela 1 – Estrutura da Ordem Jedi.**

<b>Youngling ou Iniciante Jedi</b>	São crianças sensíveis à Força que estão nos primeiros estágios do treinamento Jedi.
<b>Padawan</b>	São os aprendizes Jedi, sendo treinados por um Mestre Jedi ou um Cavaleiro Jedi até serem graduados Cavaleiros Jedi.
<b>Cavaleiro Jedi</b>	Após um Padawan ser graduado, ele se torna um Cavaleiro Jedi.
<b>Mestre Jedi</b>	Título geralmente alcançado por Cavaleiros que conseguiram fazer seus Padawans serem nomeados para Cavaleiros Jedi.
<b>Mestre Jedi Conselheiro (Consul)</b>	A estes normalmente são denominados títulos, são os que têm autoridade máxima na ordem.
<b>Grande Mestre Jedi</b>	O título dado para o líder da Ordem inteira.

**Fonte: STAR WARS WIKI (2015c, Adaptado)**

Os Jedi são os guerreiros que lutam pelo bem no universo Star Wars sendo também o foco principal dentro da série. Dentre os Jedi pode-se destacar alguns personagens muito importantes nos filmes, como Anakin Skywalker, Obi-Wan Kenobi e Mestre Yoda, ilustrados na figura a seguir.

**Figura 21 – Personagens principais dos filmes Star Wars (LUCASFILM, 2008).**



Na Figura 21 estão presentes: Anakin Skywalker (à esquerda), Obi-Wan Kenobi (ao centro) e Yoda (à direita), sendo uns dos principais personagens na trilogia inicial da saga.

### 3.1.3. Star Wars Wiki

*Wikia* (<http://www.wikia.com/>) é um portal *web* com mais de 400 comunidades criadas por fãs para fãs das mais diversas formas de entretenimento, como jogos, filmes, história em quadrinhos e livros (WIKIA, 2015). As comunidades são como enciclopédias multilíngues de licença livre, baseado na *web* e escritas de maneira colaborativa pelos usuários, funcionando assim, semelhantemente ao *Wikipédia*.

Dentre as comunidades do *Wikia*, existem 37 comunidades voltadas para o universo *Star Wars*, cada uma em um idioma específico. A *Star Wars Wiki* (<http://pt.starwars.wikia.com/>) é a comunidade em português, rica de informações do universo *Stars Wars*, incluindo quadrinhos, filmes e séries de TV. Essa comunidade foi criada baseando-se na comunidade *Wookieepedia* (<http://starwars.wikia.com/>), que conta com um acervo com mais de 119 mil artigos (WOOKIEEPEDIA, 2015). A *Star Wars Wiki* contém mais de 4.700 artigos em português sobre o universo *Star Wars* (STAR WARS WIKI, 2015). A Figura 22 ilustra a página inicial da comunidade *Star Wars Wiki*.

**Figura 22 – Página principal da comunidade *Star Wars Wiki* (STAR WARS WIKI, 2015d).**

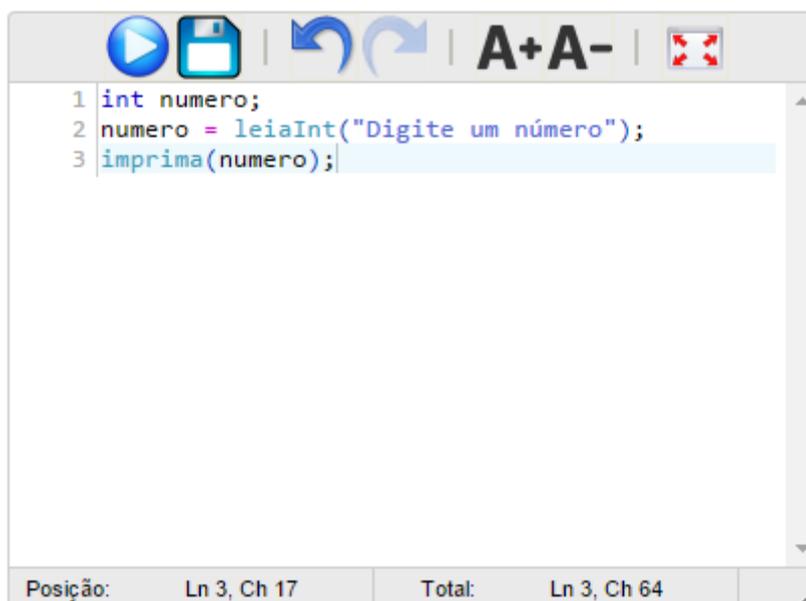
The image shows the main page of the Star Wars Wiki in Portuguese. At the top, there is a search bar and a 'Sign in' button. Below that, a navigation bar includes 'Pela Wiki', 'Eras', 'Jedi', 'Sith', and 'Star Wars Wiki'. A secondary navigation bar lists 'Atividade na Wiki', 'Página aleatória', 'Vídeos', 'Fotos', 'Chat', and 'Maps'. The main content area starts with 'Página inicial' and a social media share icon. A large blue box contains a welcome message: 'Bem-vindo à Star Wars Wiki em Português, ...uma enciclopédia de Star Wars que você mesmo pode editar, totalmente em português. Existe um número superior a 4 761 artigos em nossos arquivos desde sua inauguração, em 3 de Maio de 2006.' Below this is a warning: 'Aviso: Esta wiki contém spoilers. Leia sob sua responsabilidade.' The page is organized into four main sections: 'Navegação' (Navigation) with a call to action for help, 'Novo?' (New?) with a registration prompt, 'Artigo destacado' (Featured article) highlighting the 'Império Infinito', and 'Os filmes' (The films) listing the first seven movies of the saga.

Na Figura 22 é apresentada a página principal do *Star Wars Wikia* que é composta por diversas seções com informações e links importantes, dentre eles:

- Um campo de busca de artigos;
- Um menu para acesso aos principais temas da saga;
- Um artigo em destaque;
- Curiosidades;
- Links para os artigos dos nove filmes da franquia, os três últimos ainda não foram lançados; e
- Links para as categorias do universo *Star Wars*.

### 3.1.4. JLive 2

JLive 2 é um projeto de desenvolvimento de um ambiente de programação totalmente online feito pelo acadêmico e autor deste trabalho, Djonathas Cardoso, como trabalho de estágio curricular no ano/semestre 2014/1, orientado pelo professor Jackson Gomes (CARDOSO; GOMES, 2014). Esse ambiente permite a codificação e execução de algoritmos construídos a partir de uma linguagem semelhante ao Java, porém mais simplificada e com alguns recursos em português. A figura a seguir ilustra o ambiente do JLive 2.

**Figura 23 – Tela do editor de código do JLive2**

```
1 int numero;  
2 numero = leiaInt("Digite um número");  
3 imprima(numero);
```

Posição: Ln 3, Ch 17      Total: Ln 3, Ch 64

Conforme mostra a Figura 23, o código ilustrado possui o uso de dois recursos não pertencentes ao Java original, os métodos “leiaInt” e “imprima”. O primeiro serve para solicitar uma entrada de dados por parte do usuário, neste caso específico, um dado do tipo “int”. Já o segundo é um comando para mostrar na tela o conteúdo recebido por parâmetro. As funcionalidades dos dois comandos existem no Java, porém com uma sintaxe diferente e mais complexa da usada no JLive 2, que tem como um dos seus objetivos simplificar a linguagem Java.

Sendo uma plataforma totalmente web, o JLive 2 possui recursos que permitem o seu uso sem a necessidade de nenhum tipo de instalação por parte do usuário e com a possibilidade de ser acessível de qualquer lugar.

### 3.2. Métodos

O desenvolvimento do presente trabalho foi dividido em duas fases:

1. Levantamento do referencial teórico sobre os conceitos relacionados ao trabalho, como jogos e gamificação; e
2. Desenvolvimento do ambiente proposto.

A primeira fase, realizada no segundo semestre de 2014, consistiu na elaboração do projeto de pesquisa, incluindo toda a revisão de literatura, para então, no primeiro semestre de 2015 ser desenvolvida a segunda fase.

O desenvolvimento da segunda fase seguiu as seis etapas definidas por Werbach e Hunter (2012), o *framework* D6, para a obtenção de uma gamificação eficiente:

1. Definir Objetivos de Negócio;
2. Delimitar Comportamentos Alvo;
3. Descrever Jogadores;
4. Planejar Ciclos de Atividades;
5. Garantir Diversão; e
6. Implantar Ferramentas Apropriadas.

As seções seguintes descrevem como estas etapas foram utilizadas para guiar o desenvolvimento do trabalho.

### **3.2.1. Definir objetivos do negócio**

A primeira etapa constituiu na definição dos objetivos do negócio, em outras palavras, essa etapa definiu a finalidade e as fronteiras do sistema no projeto, semelhante à etapa de Levantamento de Requisitos, comumente utilizada em metodologias de desenvolvimento de software, tendo como diferencial o planejamento da gamificação.

Os objetivos do negócio foram definidos a partir da premissa que deveria ser obtida ou construída em um ambiente virtual de programação e que este precisaria ser gamificado. Não obstante, a experiência própria do autor do trabalho como aluno de computação foi uma contribuição importante no processo de identificação e definição dos objetivos desse ambiente.

Assim que concluída a fase de definição dos objetivos, uma pergunta precisou ser respondida: “Por que gamificar este ambiente?”. O domínio da programação exige bastante prática que por sua vez exige concentração e motivação. Os jogos conseguem manter os jogadores concentrados em uma atividade durante horas, apenas para vencer um amigo, ultrapassar desafios e descobrir o fim de uma história. Com isso, trazendo os elementos dos jogos ao contexto do sistema, através da gamificação, também possibilitaria conseguir tais benefícios.

A partir da confirmação do uso da gamificação no projeto, foram então redefinidos e expandidos os objetivos de negócio, desta vez aplicando os elementos de gamificação, em outras palavras, foi aplicada a gamificação aos objetivos de negócio.

Werbach e Hunter (2012) afirmam a necessidade de classificar a importância de cada objetivo de negócio, como uma forma de definir prioridades no projeto. Desta forma, os objetivos foram classificados a partir da sua prioridade no projeto, podendo ser normal, média ou alta.

Um elemento adicional ao escopo original do projeto foi utilizado para amparar as estratégias de gamificação e a construção do próprio ambiente: a utilização do universo *Star Wars* como metáfora. Como o uso de uma metáfora interfere em todo o desenvolvimento da dinâmica do ambiente ela foi definida nesta primeira etapa. Assim, a definição dos detalhes dos objetivos do negócio foi guiada pela metáfora. Por exemplo, elementos do universo *Star Wars* foram utilizados para definir classes de jogadores e as regras de interação entre elas. A seção 4.6.2 descreve detalhadamente a utilização dos elementos do universo *Star Wars*.

Com todos os objetivos bem definidos, a próxima etapa contempla a delimitação dos comportamentos dos objetivos anteriormente definidos. Essa etapa é descrita na subseção a seguir.

### **3.2.2. Delimitar Comportamentos Alvo**

Conforme a etapa anterior, os objetivos do negócio foram definidos a partir do problema e das hipóteses apresentadas na primeira fase deste trabalho. Os objetivos compõem os elementos essenciais do projeto, mas eles não especificam o funcionamento e as regras de interação com o jogador. Os comportamentos têm como finalidade especificar e expandir os conceitos dos objetivos, levando em consideração as características básicas de um sistema, como: interações e usabilidade.

Portanto, na etapa Delimitar Comportamentos Alvo foram identificados os possíveis comportamentos do ambiente e, então, delimitadas as regras que cada um possa ter. Alguns comportamentos necessitavam do uso de métricas para auxiliar na sua finalidade. As métricas são como uma unidade de medida utilizada no auxílio ou na complementação do comportamento, por exemplo: em um comportamento que define o mecanismo de vitória de um desafio, uma métrica de tempo pode ser usada como critério de desempate.

Por se tratar de um ambiente gamificado, foi necessário também adicionar alguns elementos de gamificação nos comportamentos, de forma a complementar sua finalidade. Por exemplo, levando em consideração o exemplo anterior do mecanismo

de vitória, pode-se inserir “recompensas” como elemento de gamificação neste comportamento.

Na próxima etapa descrever mais especificamente os jogadores do ambiente.

### **3.2.3. Descrever Jogadores**

Na etapa Descrever Jogadores foi feita a identificação e descrição dos possíveis segmentos de jogadores do Code Live. Segundo Werbach e Hunter (2012), nem todos os jogadores são iguais, logo identificar os prováveis perfis de jogadores pode contribuir no rastreamento de informações sobre comportamento, preferências e usabilidade do ambiente.

Os perfis ou grupos de jogadores foram definidos levando em consideração o foco do ambiente que é a prática de programação, mais especificamente nos níveis iniciais do aprendizado (lógica da programação).

### **3.2.4. Planejar Ciclos de Atividades**

Segundo Werbach e Hunter (2012), existem dois tipos de ciclos de atividades: Laços Interligados e Escadas de Progressão. A etapa Planejar Ciclos de Atividades consistiu em identificar e planejar os ciclos de atividades do Code Live. A definição dos ciclos depende muito das características que compõem o ambiente, por exemplo, se o ambiente for ter progressão de níveis, geralmente usa-se um ciclo de escadas de progressão.

Uma vez que foi decidido que o Code Live teria um ciclo para progressão de níveis de experiência foi necessário criar um sistema de níveis, onde para progredir de um nível a outro é necessário obter pontos de experiência. Cada progressão de nível exige uma quantidade de pontos diferente, quanto maior o nível em questão, maior será a quantidade necessária de pontos.

Pelo modelo de escadas de progressão, a evolução de nível deve ocorrer mais rapidamente nos primeiros níveis, ficando mais lenta ao alcançar os níveis mais altos.

Um sistema pode ter diversos ciclos em seu escopo, inclusive de ambos os tipos. Para que possam ser definidos os ciclos com mais exatidão, é necessário obter os resultados das etapas anteriores, de forma a ter o máximo possível do contexto do ambiente pronto, assim torna-se possível realizar o planejamento dos ciclos.

### 3.2.5. Garantir Diversão

Conforme Houaiss (2001) definiu, um jogo é uma “atividade cuja natureza ou finalidade é a diversão”. Da mesma forma, a gamificação sendo um processo para trazer elementos de jogos a outros ambientes, tem a mesma finalidade dos jogos, a diversão. Para Kuutti (2013) a “diversão é o núcleo dos jogos e gamificação” logo a sua ausência pode se caracterizar como a perda da identidade de ambiente gamificado. Na etapa Garantir Diversão uma pergunta muito importante precisou ser respondida sobre o Code Live: “ele é divertido?”.

Quando há diversão no sistema, os jogadores serão mais propensos a retornarem e jogarem novamente, já que a diversão causa também um efeito motivador. A diversão está diretamente relacionada ao sentimento de alegria, desta forma se o jogador se sente feliz dentro do ambiente é possível que esteja se divertindo também. Mas como medir a “diversão” de um sistema que ainda não está em execução ou nem sequer foi implementado?

O desafio nesta etapa foi identificar quais elementos da gamificação poderiam tanto ser divertidos quanto adaptáveis ao conceito do ambiente de programação. Para tanto, foram feitas observações em outros *cases* de sucesso, como outros ambientes gamificados e até mesmo alguns jogos.

### 3.2.6. Implantar Ferramentas Apropriadas

A etapa de Implantação das Ferramentas Apropriadas é também a última fase do *framework* D6 e também a fase na qual, de fato, ocorrem as implementações da gamificação.

Nas etapas anteriores, alguns elementos de gamificação chegaram a ser definidos, mas nesta etapa foi onde se definiu a lista completa dos elementos e como seriam usados dentro do ambiente. Os elementos foram escolhidos levando em consideração os resultados das etapas anteriores.

Na primeira etapa do *framework* D6 foi definido que haveria o uso de uma metáfora. Também foi definido que essa metáfora seria o universo *Star Wars*. No entanto, a primeira etapa só contemplou a definição inicial da metáfora, ou seja, somente a escolha do tema da metáfora e a busca por uma fonte de conteúdo sobre ela.

Nesta sexta etapa foi, então, definido o que seria usado da metáfora e onde seria usado. Como a metáfora se liga diretamente aos elementos de gamificação do ambiente, foram feitas as devidas descrições desses elementos em conjunto.

Para melhor organizar os elementos, tanto os de gamificação quanto os do universo *Star Wars*, e o funcionamento da gamificação foi realizada a confecção de um *game design*. No *game design* são criados e planejados os elementos, regras e dinâmicas de um jogo.

Com o *Game Design* definido, a próxima etapa é a codificação. Este projeto foi codificado em nível de protótipo. O protótipo fornece a camada de interface do *software* final, além de também contar com uma pequena parte da camada de processamento. O protótipo foi desenvolvido utilizando exclusivamente páginas HTML com auxílio de arquivos CSS e *Javascript*. Também foi aplicado o *framework AngularJS* que contribuiu significativamente na implementação da camada básica de processamento.

## 4 RESULTADOS E DISCUSSÃO

Para solucionar o problema proposto neste trabalho definiu-se como objetivo a criação de um ambiente virtual de programação com recursos de gamificação. A construção desse ambiente, denominado *Code Live*, seguiu o framework D6 como metodologia de desenvolvimento. As próximas subseções apresentaram os resultados obtidos durante cada etapa do desenvolvimento deste trabalho.

### 4.1. Definir Objetivos de Negócio

Como resultados da etapa Definir Objetivos de Negócio, foram identificados três objetivos:

1. Qualquer pessoa poderá se cadastrar e interagir com o ambiente;
2. O ambiente deve conter conteúdo que contribua na prática de programação;
3. O ambiente deve ser colaborativo, permitindo que os usuários criem novos conteúdos.

Considerando que na definição do projeto estava previsto o uso da gamificação, foram então redefinidos e expandidos os objetivos de negócio, desta vez aplicando os elementos de gamificação, o resultado foi uma lista com quatro objetivos:

1. Qualquer pessoa poderá se cadastrar e interagir com o ambiente como um jogador;
2. O ambiente deve conter desafios que contribuam na prática de programação;
3. O ambiente deve ser colaborativo, permitindo que os usuários criem novos desafios;
4. O ambiente deve proporcionar competitividade entre os jogadores.

Seguindo os passos desta primeira etapa, Werbach e Hunter (2012) dizem que é necessário classificar a importância de cada objetivo, como uma forma de definir prioridades no projeto. Desta forma, os objetivos foram classificados em prioridade normal, média e alta:

- **Objetivo 1 – Prioridade normal**

- **Objetivo 2 – Prioridade alta**
- **Objetivo 3 – Prioridade média**
- **Objetivo 4 – Prioridade média**

O objetivo 1 garante que qualquer indivíduo tenha acesso ao ambiente, independente se cursa ou não algum curso relacionado a programação. Além disso, o cadastro do jogador permite que sejam implementados diversos elementos, como pontuação e moeda, já que esses precisam persistir de forma individual para cada jogador. Este objetivo foi classificado como prioridade normal pois a ausência de um cadastro de jogador não impediria que alguém jogasse o jogo, no entanto, não seria possível utilizar alguns elementos que exigem ser armazenados, como citado anteriormente.

O objetivo 2 é considerado o mais importante, já que os desafios são a base de toda a dinâmica do sistema. É a partir dos desafios que deve ser obtida a prática da programação, por esse motivo sua prioridade foi dada como alta.

O objetivo 3 aborda o quesito colaborativo do ambiente, que seria a possibilidade de os jogadores criarem novos desafios. Esta possibilidade aos jogadores é interessante para trazer uma outra dimensão do treinamento da programação: a criatividade. Outro ponto é que a criação exige mais raciocínio lógico do jogador, sendo assim, aqueles que têm habilidade na criação teriam menos dificuldade na resolução. O recurso também contribui para o crescimento do ambiente, já que se houver constante inserção de novos desafios aumenta-se as opções para os outros jogadores e evita que o ambiente fique defasado. Por ser um objetivo que contribui bastante com o ambiente e sua ausência não evita seu funcionamento normal, ele foi classificado como prioridade média.

O quarto e último objetivo fala sobre competitividade entre jogadores. A competição é um elemento previsto na gamificação e é essencial para trazer mais motivação e permanência em um jogo, já que sempre um jogador tenta ser melhor que o outro, desta forma a competição nunca termina. Foi dado como prioridade média por não impedir o funcionamento básico do ambiente.

Como definido nos métodos do projeto, também foi escolhida a metáfora do ambiente. Então foi decidido que o Code Live utilizaria o universo dos filmes “*Star Wars*”, para contribuir na definição de algumas dinâmicas e comportamentos do ambiente.

Uma vez decidido o tema da metáfora, foram realizadas pesquisas para encontrar conteúdo que pudesse ser utilizado como embasamento para os elementos de gamificação que seriam implementados. Dentre os diversos sites encontrados, o que mais chamou a atenção, devido a sua riqueza e detalhes sobre a série, foi o **Star Wars Wiki**.

Com todos os objetivos bem definidos, a próxima etapa contempla a delimitação dos comportamentos dos objetivos anteriormente definidos. Essa etapa é descrita na subseção a seguir.

#### **4.2. Delimitar Comportamentos Alvo**

Com os objetivos prontos, nesta segunda etapa foram delimitados os comportamentos alvo, ou seja, foi definido o que os jogadores podem fazer no ambiente e como isso é medido. A partir dos quatro objetivos definidos na etapa anterior foram definidos comportamentos e sobre alguns desses, foram aplicadas métricas e conceitos de gamificação. Para melhor organização, os comportamentos foram categorizados em três categorias de acordo com o seu contexto:

##### **C1 - Jogador**

**C1.1** - Qualquer pessoa pode se cadastrar como usuário preenchendo um formulário de cadastro com os seguintes campos: Nome completo, Endereço de E-mail e Nome de Exibição.

**C1.2** - Todas as informações fornecidas pelo usuário podem posteriormente serem editadas, com exceção do “Nome de exibição”.

**C1.3** - Um jogador pode selecionar um avatar para o representá-lo.

##### **C2 - Desafios**

**C2.1** – Um jogador pode criar e publicar novos desafios;

**C2.2** – Enquanto não publicado, o desafio pode ser editado ou excluído, após isso, apenas desativado;

**C2.3** – Um desafio deve ter um nome (que deverá ser único no ambiente), descrição, tags, um enunciado, um objeto/vetor de entrada, objeto/vetor de saída;

- **Métricas:** Um tempo máximo para resolução e recompensas (moeda e experiência);

**C2.4** – Desafios criados por jogadores inexperientes devem ser validados antes de serem disponibilizados ao público;

- **Métricas:** Basear a in experiência de acordo com o nível de experiência do jogador;

**C2.5** – Os desafios podem ser julgados pelos jogadores com notas que irão variar de 1 a 5, sendo 1 considerada uma nota péssima e 5 uma nota ótima;

**C2.6** – Desafios que não são claros no seu objetivo ou irresolvíveis podem ser relatados por outros jogadores. As reclamações são avaliadas por moderadores e caso procedente, o desafio deve ser desativado;

**C2.7** – O jogador tem uma tela de acesso com o histórico dos seus desafios e reclamações de outros jogadores sobre os desafios de sua autoria;

**C2.8** – Os jogadores podem participar de desafios de outros jogadores, mas não os de sua própria autoria;

- **Métricas:** Adicionar requisitos para participação, como custo em moeda e nível mínimo do jogador;

**C2.9** – Uma lista com todos os desafios publicados deve ficar disponível aos jogadores;

- **Métricas:** Classificar desafios em níveis de dificuldade e colocar requisitos de participação de acordo com cada nível;

**C2.10** – Na lista de desafios somente informações básicas sobre o mesmo devem ficar disponíveis, como descrição, tags, nota média das avaliações e nome do autor;

**C2.11** – A resposta de um desafio deve ser validada através da comparação das saídas do jogador com as saídas esperadas (descritas na criação do desafio). Somente se todas as saídas foram compatíveis o desafio é dado como resolvido;

- **Métricas:** Gerar recompensas em moeda e experiência quando dado como resolvido o desafio.

**C2.12** – Quando um jogador vence um desafio, este não poderá ser resolvido novamente por ele. Já quando houver derrota, o jogador se desejar realizar o desafio novamente poderá comprá-lo refazê-lo completamente.

### **C3 – Dinâmica Básica**

**C3.1** – A experiência gera a progressão de níveis e classes do jogador.

- **Métricas:** A experiência do jogador é medida em pontos chamados de Domínio da Força (DF).

Como pode ser percebido, os comportamentos se assemelham com o levantamento de requisitos de um software, já que ambos buscam delimitar as

fronteiras do sistema. Já a próxima etapa descreve mais especificamente os jogadores do ambiente.

#### 4.2.1. Descrever Jogadores

Na etapa 3 foi feita a identificação e descrição dos possíveis segmentos de jogadores do Code Live. Segundo Werbach e Hunter (2012), nem todos os jogadores são iguais, logo identificar os prováveis perfis de jogadores pode contribuir no rastreamento de informações sobre comportamento, preferências e usabilidade do ambiente.

Levando em consideração as características do ambiente, espera-se quatro grupos distintos de jogadores, sendo eles:

1. Alunos;
2. Professores;
3. Curiosos; e
4. Entusiastas.

Como o foco do ambiente é a prática de programação, mais especificamente nos níveis iniciais do aprendizado (lógica da programação), um dos possíveis grupos de jogadores é formado por **alunos** que cursam disciplinas iniciais de programação.

Espera-se também que haja um grupo de **professores** dentro do ambiente, já que existe a possibilidade de criar novos desafios para os jogadores, tornando assim até uma forma de atividade complementar para uso dos professores com os alunos.

A programação tem se tornado mais visível na mídia. Filmes como “A rede social” (2010) e “Os estagiários” (2013) mostram a programação de forma positiva e empolgante fazendo com que surjam **curiosos** interessados em conhecer essa ciência. Estes indivíduos podem sentir-se motivados a ingressar em um curso de programação, mas antes buscarão formas de “conhecer” a programação para então tomar uma decisão definitiva.

O último grupo esperado é formado por indivíduos que são **entusiastas** quando o assunto é programação. Alguns são motivados pelo trabalho, já outras programam por “diversão”.

A possibilidade da existência de professores além de garantir e intensificar a inclusão do recurso de criação de desafios, também foi incluído um recurso que permite que os professores cadastrados possam criar desafios sem passar por validação para serem publicados. Isso garante mais agilidade para esse grupo de usuários que devem focar nesse recurso do ambiente.

Por último, um pequeno grupo que também pode surgir como jogador são os chamados “curiosos”. Esses jogadores ingressariam sem habilidade em programação, o que levaria a possibilidade dos mesmos não conseguirem realizar os desafios do Code Live, baseando-se nos moldes atuais do ambiente. Isso acontece devido à ausência de recursos efetivos de aprendizagem para esse nível de jogador. Como um trabalho futuro, prevê-se adicionar recursos que proporcionem “aprendizagem” ao ambiente. A próxima etapa irá descrever os ciclos de atividades do ambiente.

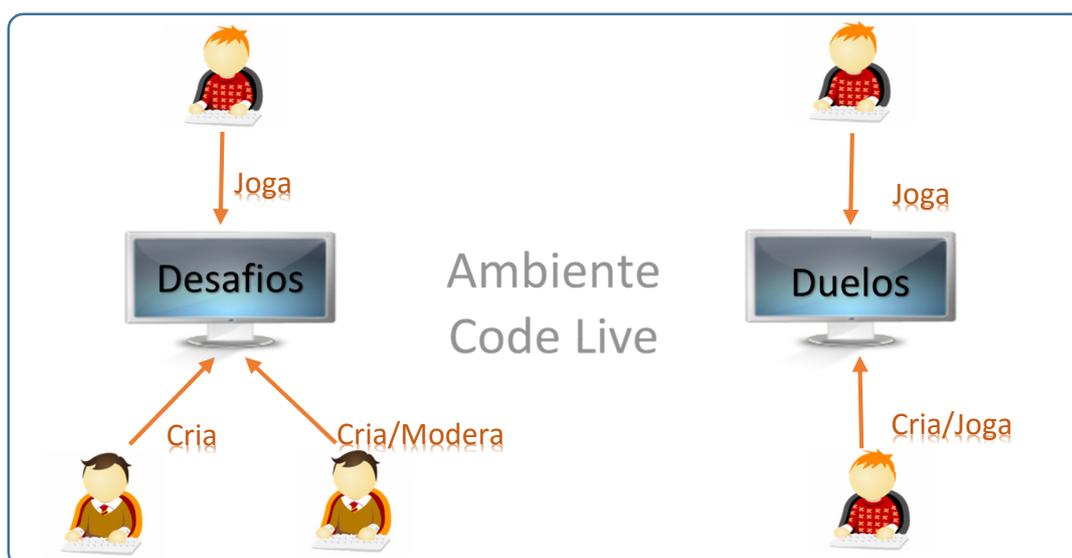
### 4.3. Planejar Ciclos de Atividades

Foram planejados dois ciclos, cada um utilizando um dos dois tipos de atividades. O ciclo principal, também denominado neste projeto como **ciclo dos desafios**, é um ciclo do tipo laço interligado que abrange a jogabilidade do ambiente. O outro ciclo presente no ambiente é conhecido como **ciclo de evolução**, neste ciclo é mostrado o funcionamento da progressão de níveis de experiência do jogador. Esses dois ciclos existentes no Code Live serão apresentados nas subseções a seguir.

#### 4.3.1. Ciclo de Desafios

O ciclo de desafios compõe o ambiente básico do *Code Live*, onde se encontram as principais interações e dinâmicas do ambiente. A figura a seguir ilustra conceitualmente esse ambiente.

**Figura 24 – Conceito básico do ambiente do Code Live**



Desta forma, conforme mostra a Figura 24, há existência também de dois tipos de atores no ambiente, cada um com funções específicas. O ator “jogador” joga os

desafios, enquanto o ator “mestre” cria desafios para o ambiente. O ator “mestre” modera desafios criados por outros “jogadores”. Já em relação aos duelos, os dois atores podem realizar a tarefa de criar um duelo e outro de ser desafiado, ambos jogam sobre as mesmas regras.

A composição do ciclo de desafios segue a estrutura de um ciclo do tipo laço interligado, conforme o *framework* D6, tendo três elementos interligados: motivação, ações e feedback. A figura a seguir ilustra a composição do ciclo dos desafios como laço interligado.

**Figura 25 – Ciclo de Desafios do Code Live**



Como pode-se observar na Figura 25, o fato de o ciclo ser classificado como um laço não é ao acaso, a intenção é que realmente seja um ciclo contínuo que incentive o jogador a permanecer ativo no Code Live.

O elemento **motivação**, do ciclo de desafios, é o incentivo para a realização das **ações** que, quando concluídas com sucesso, geram **feedback**. O feedback compreende exatamente o que foi almejado na motivação.

Foram encontradas três **motivações** nesse ciclo principal do Code Live, sendo elas: ganho de créditos, progressão de nível e ranking. Os créditos são requisito para compra de desafios e auxílios dentro de um desafio, logo a ausência de créditos limita o jogador, essa limitação gera motivação para obtenção de mais créditos. O ganho de experiência também entra neste ciclo na forma de motivação por ser a única forma de progressão de nível, assunto que será abordado mais adiante na descrição do ciclo de evolução, e também por ser a única forma de ganhar posições no ranking, sendo

este o último item da motivação, já que ele proporciona a competitividade entre os jogadores.

Já as **ações** que compõem o ciclo de desafio também são três: resolver desafio, resolver duelo e criar desafio. Um **desafio** é um exercício de programação composto por um enunciado que descreve um problema a ser solucionado pelo jogador através de um algoritmo que ele criará. Um **duelo** funciona semelhantemente ao desafio, porém, no duelo, dois jogadores tentam solucionar um mesmo desafio, o que conseguir fazê-lo em menos tempo, vence. A **criação** de um desafio resume-se em criar um desafio que posteriormente poderá ser jogado por outros jogadores na forma de desafio ou duelo. Essas três ações funcionarão de forma independente, ou seja, não é necessário que o jogador execute as três ações para alcançar o feedback. Todas as motivações anteriormente mencionadas também se enquadram com todas as ações de forma independente.

Por fim, quando uma das ações anteriores é completada com sucesso é gerado ao jogador um **feedback**. Feedback é o reconhecimento do sucesso do jogador na resolução do desafio ou duelo, reconhecimento dado em forma de “recompensa”. As recompensas, que serão explicadas com mais detalhes mais adiante no trabalho, são compostas de uma quantidade de créditos e de experiência, que se encaixa com as motivações. Desta forma fecha-se o ciclo, pois mesmo a recompensa gerando ganho ao jogador, sempre haverá a necessidade de crescer ainda mais e ficar à frente dos outros jogadores, criando assim novamente motivação, dando continuidade ao ciclo.

#### **4.3.2. Ciclo de Evolução**

O ciclo de evolução é a definição do funcionamento do sistema de progressão de níveis de experiência do jogador. Seguindo o modelo das escadas de progressão, este ciclo em questão é composto por dezoito níveis de experiência, sendo que para progredir de um nível a outro é necessário obter Domínio da Força (DF). Cada progressão de nível exige uma quantidade de pontos diferente, quanto maior o nível em questão, maior será a quantidade necessária de pontos.

A definição mais detalhada da progressão de níveis está prevista no Game Design deste trabalho (seção 4.5.3.1), a ser apresentado mais adiante.

#### 4.4. Garantir Diversão

Na etapa 5 uma pergunta muito importante precisou ser respondida sobre o Code Live: ele é divertido? Mas como medir a “diversão” de um sistema que ainda não está em execução ou nem sequer foi implementado? Para solucionar esse questionamento foram avaliados os elementos que poderiam gerar divertimento entre os jogadores:

- Níveis de experiência do jogador;
- Níveis de dificuldade de desafio progressivos;
- Acúmulo de moeda virtual; e
- Competição com outros jogadores.

Esses quatro elementos também fazem parte da lista dos elementos da gamificação, definidos por Kuutri (2013), e que serão explicados em uma seção dedicada mais adiante (seção 4.5.1).

Os níveis de experiência do jogador são uma forma de medir o nível de desenvolvimento e habilidade do jogador dentro do ambiente. Quando um jogador “sobe” de nível, um sentimento de alegria pode ser sentido pelo jogador, já que esse evento demonstra ao jogador que o mesmo está crescendo dentro do ambiente.

A mudança de nível de experiência ocasiona também o aparecimento de outro elemento de diversão: a progressão nos níveis de dificuldade dos desafios. Os desafios são divididos em níveis de dificuldade, cada nível possui características que o fazem ser mais complexo ou não que outro. Para acessar determinados níveis de dificuldade de desafio é necessário que o jogador tenha um nível mínimo de experiência. Desta forma quando o jogador alcança um certo nível de experiência, além do sentimento de realização por essa conquista, também terá um sentimento competidor para enfrentar os próximos desafios de grau maior de complexidade.

O acúmulo ou ganho de moeda proporciona ao jogador benefícios, já que a mesma é requisito para ações dentro do ambiente. Funciona semelhantemente a vida real, quanto maior o dinheiro, mais possibilidade de acesso a recursos se tem. Logo o ganho de moeda irá proporcionar diversão ao jogador não só pelo fato de receber a moeda, mas pelo que se poderá fazer com ela.

Jogar sozinho pode ser muito tedioso e pouco desafiador, por isso, um dos elementos que devem trazer maior diversão entre todos os tipos de jogadores é a competição multijogador. No âmbito do Code Live essa competição se dá na forma de duelos entre dois jogadores, como apresentado anteriormente nos ciclos de

atividades. Outro ponto importante desse elemento é que ele proporciona o ganho dos outros três elementos de diversão apresentados anteriormente simultaneamente.

#### **4.5. Implantar Ferramentas Apropriadas**

Na sexta e última fase do *framework* D6 tem-se início o planejamento do que será de fato implementado para o ambiente. Nesta fase foram definidos diversos componentes essenciais ao ambiente do Code Live, sendo eles:

- Elementos de Gamificação;
- Elementos do Universo *Star Wars*; e
- *Game Design*

Estes componentes são descritos nas seções a seguir.

##### **4.5.1. Elementos de Gamificação**

A escolha e definição dos elementos de gamificação que foram utilizados na implementação do projeto foram baseadas na experiência de diversas outras ferramentas que tem como objetivo semelhante ao Code Live, como o Code Hunt (seção 2.3.2) e o Code School (seção 2.3.3). A lista a seguir apresenta a relação dos elementos escolhidos de acordo com sua categoria:

- **Dinâmicas**
  - Progressão;
  - Relacionamentos; e
  - Restrições
- **Mecânica**
  - Desafios;
  - Competição;
  - Feedback;
  - Recompensas; e
  - Transações;
- **Componentes**
  - Emblemas;
  - Quadro de Líderes;
  - Níveis; e

- Conquistas.

#### 4.5.2. Elementos do Universo *Star Wars*

Com o uso do universo *Star Wars*, foi necessário definir o que poderia de fato ser utilizado de forma eficiente no projeto. Esses elementos foram escolhidos conforme sua adequação aos elementos de gamificação definidos para o projeto. A seguir é mostrada a lista de elementos do universo *Star Wars* (SW) utilizados no ambiente:

- Estrutura da Ordem Jedi;
- Frases de ênfase que são ou que remetem ao universo SW;
- Conceitos do universo SW;
- Imagens dos personagens.

Já em relação ao uso das frases do SW, na sua grande maioria, foram formuladas a partir da junção de conceitos, personagens, comportamentos e características do universo SW. Por exemplo, o personagem “Mestre Yoda” usualmente coloca os verbos (principalmente verbos auxiliares) após o objeto e do sujeito (um formato objeto-sujeito-verbo), como na frase “Quando novecentos anos você tem, boa aparência você não tem”. Esse formato de composição de frase foi utilizado pelo ambiente para a formação de frases próprias (não originadas do SW), como por exemplo: “Um desafio para você tenho, jovem Padawan”. Também há o uso de frases elaboradas a partir de conceitos do universo SW, como: “quero ser Jedi” e “pegue seu sabre de luz”, como também de frases originadas do próprio filme, por exemplo: “a Força está comigo!” e “eu sou seu pai”.

Como descrito no parágrafo anterior, há o uso de conceitos em algumas frases do ambiente. Além do uso em frases, esses conceitos também são utilizados de forma independente (sem composição de frase).

Por fim, algumas imagens de personagens foram utilizadas. Essas imagens foram obtidas a partir da série de animação em 3D chamada *Star Wars: Clone Wars*.

Esses elementos do SW mencionados anteriormente foram utilizados na definição de algumas partes do sistema, como: níveis de experiência do jogador, níveis de dificuldade dos desafios, conquistas, entre outros e serão apresentados no decorrer deste trabalho.

### 4.5.3. Game Design

O Game Design é o projeto de toda a dinâmica e interação do ambiente, foi dividido em quatro seções que apresentam os elementos essenciais do Code Live, sendo eles: Dinâmica do Sistema, Modo Criação, Modo Desafio e Modo Duelo. A próxima seção abrange as características da Dinâmica do Sistema.

#### 4.5.3.1. Dinâmica do Sistema

Esta seção irá descrever o funcionamento da dinâmica básica do sistema Code Live. Para melhor organização, esta seção será subdividida em subseções de acordo com as temáticas específicas.

##### **Cadastro do Jogador**

Qualquer pessoa pode se cadastrar como jogador preenchendo um formulário de cadastro com os seguintes campos:

- Nome completo;
- Endereço de e-mail;
- Nome dentro do jogo (personagem).

Todas as informações fornecidas pelo jogador podem posteriormente ser editadas, com exceção do “Nome dentro do jogo”, pois causaria inconsistências com as ações já realizadas pelo jogador dentro do ambiente.

Finalizando o preenchimento do formulário de cadastro, é possível o jogador selecionar um **avatar** (imagens pré-definidas do tema SW) para o representar dentro do jogo.

##### **Dinheiro e Experiência**

Existe uma moeda dentro do Code Live chamada **Crédito Padrão Galáctico**, conforme o universo SW. Ela é utilizada como forma de pagamento e recompensa em diversos momentos dentro do jogo. Para facilitar sua representação textual, a moeda foi simplesmente chamada de “crédito” dentro do contexto gráfico. Todo jogador inicia com uma quantia de 50 créditos.

Os jogadores são classificados em **Classes** que são obtidas através de um nível de experiência chamado **Domínio da Força (DF)**. No contexto SW, a Força é uma energia que dá poderes especiais aos guerreiros Jedi. A tabela a seguir traz a divisão dos níveis de experiência e as classes baseadas no modelo hierárquico do

SW juntamente com informações sobre a transição desses níveis e a suas recompensas.

**Tabela 2 - Divisão de Classes de Experiência**

Nome da Classe	Transição de níveis	Recompensas
Novato	Nível 1 – 0 a 99 DF	+ 50 créditos
	Nível 2 – 100 a 249 DF	+ 70 créditos
	Nível 3 – 250 a 499 DF	+ 100 créditos
Padawan	Nível 1 – 500 a 799 DF	+ 120 créditos
	Nível 2 – 800 a 1199 DF	+ 140 créditos
	Nível 3 – 1.200 a 1.699 DF	+ 160 créditos
	Nível 4 – 1.700 a 2.499 DF	+ 180 créditos
	Nível 5 – 2.500 a 3.499 DF	+ 200 créditos
Cavaleiro Jedi	Nível 1 – 3.500 a 4.599 DF	+ 200 créditos
	Nível 2 – 4.600 a 5.799 DF	+ 210 créditos
	Nível 3 – 5.800 a 7.199 DF	+ 220 créditos
	Nível 4 – 7.200 a 8.799 DF	+ 230 créditos
	Nível 5 – 8.800 a 10.499 DF	+ 240 créditos
	Nível 6 – 10.500 a 12.399 DF	+ 250 créditos
	Nível 7 – 12.400 a 14.999 DF	+ 300 créditos
Mestre Jedi	Nível 1 – 15.000 a 17.999 DF	+ 500 créditos
	Nível 2 – 18.000 a 23.999 DF	+ 700 créditos
	Nível 3 – 24.000 a 29.999 DF	+ 1000 créditos
Mestre do Conselho	30.000 DF ou superior	+ 2000 créditos

A experiência dos jogadores (DF) é usada como base para a formação do ranking (quadro de líderes) do jogo. Os jogadores recebem conquistas e emblemas de acordo com as ações que realiza no jogo. A seguir a tabela da divisão das conquistas e como consegui-las.

**Tabela 3 – Conquistas do Code Live**

Categoria	Nome da Conquista	Requisitos	Recompensas
Evolução	Primeira evolução	Alcançou a classe Novato (Nível 2)	Recompensa do nível (Tabela 2)
	Aprendiz Jedi	Alcançou a classe Padawan (Nível 1)	Recompensa do nível (Tabela 2)
	Obi-Wan Kenobi está orgulhoso	Alcançou a classe Padawan (Nível 5)	Recompensa do nível (Tabela 2)
	Novo Jedi	Alcançou a classe Cavaleiro Jedi (Nível 1)	Recompensa do nível (Tabela 2)
	Quase um Anakin Skywalker	Alcançou a classe Cavaleiro Jedi (Nível 5)	Recompensa do nível (Tabela 2)

	A Força está com você!	Alcançou a classe Cavaleiro Jedi (Nível 7)	Recompensa do nível (Tabela 2)
	Um verdadeiro cavaleiro, você é	Alcançou a classe Mestre Jedi (Nível 1)	Recompensa do nível (Tabela 2)
	O mestre dos mestres	Alcançou a classe Mestre Jedi (Nível 3)	Recompensa do nível (Tabela 2)
	Superando o Mestre Yoda	Alcançou a classe Mestre do Conselho	Recompensa do nível (Tabela 2)
<b>Quantitativo</b>	Quero ser Jedi	Primeiro dia no jogo	+ 50 créditos
	Começando a batalha	Comprou o primeiro desafio no jogo	+ 10 créditos
	A Força está comigo	Venceu o primeiro desafio	+ 20 créditos
	Quero ser um mestre	Criou o primeiro desafio (publicado)	+ 50 créditos
	Pegue seu sabre de luz	Primeiro duelo com outro jogador	+ 20 créditos
	Eu sou seu pai	Venceu seu primeiro duelo	+ 50 créditos

As informações de experiência (DF) e créditos do jogador ficam sempre visíveis em todas as telas do ambiente. Já as estatísticas e conquistas são apresentadas em uma área dedicada ao perfil do jogador, sendo estas públicas a todos os jogadores, junto com as informações de experiência (DF).

### Poderes

Os Poderes são recursos ou auxílios que podem ser utilizados durante desafios ou duelos para ajudar o jogador na sua resolução. Cada poder tem um “efeito” e um custo. O efeito é uma ação que é aplicada para um determinado fim, podendo existir limitações de seu uso dependendo do contexto do seu uso. A tabela a seguir exemplifica os poderes que existem no Code Live como também suas características e custos.

**Tabela 4 – Poderes**

Nome do Poder	Efeito	Limitações	Custos
<b>Congele o tempo</b>	Congela o contador de encerramento do desafio, deixando o jogador com tempo ilimitado para resolver o desafio.	<ul style="list-style-type: none"> <li>❖ Só pode ser utilizado uma vez.</li> <li>❖ Não pode ser utilizado em duelos.</li> </ul>	50 créditos
<b>O dobro ou nada</b>	<ul style="list-style-type: none"> <li>❖ Em um desafio comum</li> </ul> Permite o jogador sortear um novo desafio de mesmo nível.	<ul style="list-style-type: none"> <li>❖ O poder age como uma nova compra de desafio ou aumento</li> </ul>	100 créditos

	<p>Caso vença, recebe a recompensa referente aos dois desafios.</p> <ul style="list-style-type: none"> <li>❖ Em um duelo Semelhante ao caso anterior, com a diferença que neste é dobrado o valor da aposta. Se o outro jogador não tiver como cobrir a nova aposta automaticamente o jogador que usou o poder ganha.</li> </ul>	<p>do valor da aposta, logo o jogador que está utilizando o poder deve ter créditos suficientes para cobrir esses valores.</p> <ul style="list-style-type: none"> <li>❖ O poder só terá efeito ao fim do tempo máximo do duelo ou até um dos jogadores ganhar.</li> </ul>	
<b>Anular poderes</b>	Anula ou bloqueia o uso de qualquer tipo de poder no duelo.		200 créditos

O ambiente do Code Live é organizado em três modos, o Modo Desafio, Modo Duelo e o Modo Criação, que serão apresentados nas seções seguintes.

#### 4.5.3.2. Modo Criação

O Modo Criação permite aos jogadores criarem e publicarem novos desafios. Enquanto não publicado, o desafio pode ser editado ou excluído, após isso, pode apenas ser desativado.

Um desafio é composto principalmente por um enunciado que descreve um problema específico que deverá ser solucionado pelo jogador através da criação de um algoritmo.

Os desafios são classificados através de um conjunto de níveis de dificuldade. Cada nível de dificuldade tem características específicas que os distinguem e permitem que haja uma divisão justa para todos as classes de jogadores, onde cada um faz o desafio de acordo com seu conhecimento obtido no jogo. A tabela a seguir ilustra a divisão dos níveis de dificuldade e suas características.

**Tabela 5 – Classificação dos níveis de dificuldade dos desafios**

Nome do Nível	Características
<b>Fundamentos da Força</b>	<ul style="list-style-type: none"> <li>➤ Declaração de variáveis</li> <li>➤ Tipos primitivos (int, double, char) e String</li> <li>➤ Entrada e saída de dados (funções de leitura e impressão)</li> </ul>
<b>Treinando com o Sabre de Luz</b>	<ul style="list-style-type: none"> <li>➤ Estruturas condicionais</li> <li>➤ Operadores lógicos</li> <li>➤ Operadores relacionais</li> <li>➤ Instrução if...else</li> </ul>
<b>Dominando a Força</b>	<ul style="list-style-type: none"> <li>➤ Estruturas de repetição</li> <li>➤ Contadores</li> </ul>

	<ul style="list-style-type: none"> <li>➤ Acumuladores</li> <li>➤ Instrução for</li> <li>➤ Instrução while</li> </ul>
<b>Controlando Mentes</b>	<ul style="list-style-type: none"> <li>➤ Laços encadeados</li> <li>➤ Matrizes unidimensionais (vetores)</li> <li>➤ Matrizes bidimensionais</li> </ul>

Ao ser criado, um desafio deve ter, obrigatoriamente, um nome (que deverá ser único no jogo), uma descrição resumida, tags, um enunciado, um tempo máximo de resolução e uma saída ou um conjunto de saídas. As saídas são os resultados esperados da compilação do código do desafio e que são utilizados na validação do mesmo. A criação de novos desafios é limitada aos jogadores de acordo com o nível do desafio a ser criado, o nível de experiência e saldo de créditos suficiente do jogador.

Jogadores com classe inferior a Mestre Jedi necessitam da aprovação de um jogador classe Mestre Jedi ou superior para que seu desafio seja publicado. Os demais têm publicação direta do desafio no jogo. Se um desafio criado por um jogador não for aprovado, os créditos gastos não são devolvidos. O custo do desafio serve como uma forma de “pagamento” ao serviço de moderação.

A criação de desafios gera recompensas de acordo com o nível de dificuldade do desafio. Há uma recompensa inicial em experiência (DF), dada ao criar o desafio e/ou ele ser aprovado, e há uma recompensa por “venda” do desafio a outros jogadores, esta última é contabilizada ao jogador uma vez por dia para reduzir o processamento no jogo. Todas essas informações são ilustradas na tabela a seguir.

**Tabela 6 – Requisitos e recompensas para criação de desafios**

Nome do Nível	Requisitos Mínimos	Recompensa Inicial	Recompensa por Venda
<b>Fundamentos da Força</b>	<ul style="list-style-type: none"> <li>❖ Padawan</li> <li>❖ 20 créditos</li> <li>❖ Aprovação de um Mestre Jedi</li> </ul>	+ 100 DF	+ 1 crédito
<b>Treinando com o Sabre de Luz</b>	<ul style="list-style-type: none"> <li>❖ Jedi</li> <li>❖ 50 créditos</li> <li>❖ Aprovação de um Mestre Jedi</li> </ul>	+ 200 DF	+ 2 créditos
<b>Dominando a Força</b>	<ul style="list-style-type: none"> <li>❖ Mestre Jedi</li> <li>❖ 100 créditos</li> </ul>	+ 400 DF	+ 5 créditos
<b>Controlando Mentes</b>	<ul style="list-style-type: none"> <li>❖ Mestre Jedi</li> <li>❖ 200 créditos</li> </ul>	+ 1000 DF	+ 10 créditos

Para que o Code Live tivesse uma consistência inicial, foi estabelecida uma quantidade mínima de desafios de cada nível de dificuldade para que os jogadores não permaneçam sem desafios até que sejam criados novos por outros jogadores. A tabela a seguir estabelece a quantidade mínima de desafios em cada nível de dificuldade deve ter no lançamento do Code Live.

**Tabela 7 – Quantidade mínima de desafios por nível de dificuldade**

<b>Nome do Nível</b>	<b>Quantidade Mínima de Desafios</b>
<b>Fundamentos da Força</b>	❖ 20 desafios
<b>Treinando com o Sabre de Luz</b>	❖ 10 desafios
<b>Dominando a Força</b>	❖ 5 desafios
<b>Controlando Mentes</b>	❖ 2 desafios

Os desafios podem ser julgados pelos jogadores com notas que variam de 1 a 5, sendo 1 considerada uma nota péssima e 5 uma nota ótima.

Caso um jogador encontre um problema no desafio pode relatá-lo ao sistema, sendo visível somente ao criador e aos moderadores (Mestre Jedi ou superior). As reclamações podem ser avaliadas pelos moderadores e caso seja verificado que o desafio não é claro no seu objetivo ou é irresolúvel, este pode vir a ser desativado pela moderação. O criador do desafio também pode acessar as reclamações e desativar o desafio.

O jogador tem uma tela de acesso com o histórico dos seus desafios, tão como o quantitativo de “compras” e classificação geral. Também tem acesso as reclamações dos jogadores em relação ao desafio.

Os Mestres Jedi e Mestres do Conselho atuam como moderadores dentro do jogo, além de usufruírem das mesmas funcionalidades dos demais jogadores, eles podem criar desafios sem a necessidade de passar por moderação.

Os administradores do sistema podem eleger jogadores como Mestres Jedi mesmo que este ainda não tenha o DF necessário. Neste caso, o seu DF é automaticamente elevado ao correspondente ao nível 1 do Mestre Jedi.

#### **4.5.3.3. Modo Desafio**

O Modo Desafio permite aos jogadores jogarem desafios já criados. Para jogar um desafio, o jogador deve selecionar um desafio na lista de desafios e depois “comprá-lo”. Quando um desafio é selecionado é apresentado as informações do desafio, como a descrição básica, as tags, a nota média das avaliações de outros

jogadores e o nome do autor do desafio. O enunciado não é visível para evitar que isso influencie na escolha do jogador.

Cada desafio possui um nível de dificuldade, que por sua vez possui requisitos para ser jogado. Quando o jogador não atende esses requisitos fica impossibilitado de realizar tal desafio.

Os requisitos para jogar se baseiam de acordo com o nível de dificuldade do desafio, o nível de experiência e o saldo de créditos disponível do jogador, conforme a tabela a seguir.

**Tabela 8 – Requisitos e recompensas para jogar desafios**

Nome do Nível	Requisitos Mínimos	Recompensas
<b>Fundamentos da Força</b>	❖ Novato ❖ 5 créditos	+ 50 DF + 15 créditos
<b>Treinando com o Sabre de Luz</b>	❖ Novato (Nível 3) ❖ 15 créditos	+ 100 DF + 50 créditos
<b>Dominando a Força</b>	❖ Padawan ❖ 25 créditos	+ 200 DF + 100 créditos
<b>Controlando Mentes</b>	❖ Jedi ❖ 50 créditos	+ 500 DF + 200 créditos

Quando um desafio é comprado, uma contagem regressiva é iniciada de acordo com o tempo máximo configurado para o desafio. Se a contagem regressiva chegar a 0 e o jogador não tiver conseguido resolver o desafio, este muda seu status para fechado, impossibilitando o jogador de continuar jogando. Nenhuma recompensa ou devolução ocorrerá neste caso.

Se o jogador desejar desistir ele pode fazê-lo acionando a opção “Desistir” ou apenas aguardando o término do tempo de resolução. Nenhuma recompensa ou devolução ocorrerá neste caso.

Para vencer um desafio, o jogador necessita resolver o problema do enunciado do desafio corretamente dentro do tempo máximo estipulado. O término do tempo máximo resulta em derrota.

#### **4.5.3.4. Modo Duelo**

O Modo Duelo permite os jogadores criarem e jogarem duelos. Um duelo é um desafio entre dois jogadores, onde somente um vence ou os dois perdem. Funciona praticamente nos mesmos moldes dos desafios comuns, porém com o diferencial que

vence quem resolver o desafio corretamente em menos tempo e que estes possuem um sistema de recompensas diferenciado.

Quando um duelo é criado, o jogador desafiado tem um prazo máximo de três dias para aceitar o desafio. Se o jogador desafiado não aceitar ou perder o prazo do duelo, o sistema dará vitória automática ao jogador que criou o desafio, assim que o mesmo encerrar corretamente o desafio.

Após iniciado o duelo, se um dos jogadores desistir, o outro vence automaticamente, assim que o mesmo encerrar corretamente o desafio. Não é necessário que os jogadores joguem simultaneamente.

Os duelos têm um prazo máximo de 7 dias para serem concluídos. Se ambos os jogadores não concluírem o desafio dentro do tempo máximo acarretará na derrota dos dois jogadores, sem direito a qualquer tipo de recompensa ou devolução.

Os requisitos de nível de experiência (DP) para criar ou entrar em um duelo são os mesmos dos desafios comuns, porém o custo em créditos e as recompensas são diferenciadas.

O custo em créditos para criar ou entrar em um duelo é dado no formato de apostas. O jogador que está criando o duelo pode definir um valor de aposta, que é determinado em intervalos de 50 em 50 créditos, ex: 50, 100, 150, [...] créditos. O valor mínimo e máximo das apostas é definido de acordo com a classe do jogador que está criando o duelo, conforme a tabela a seguir.

**Tabela 9 – Valores limites das apostas de acordo**

<b>Classe do Jogador</b>	<b>Aposta Mínima</b>	<b>Aposta Máxima</b>
<b>Novato</b>	50 créditos	250 créditos
<b>Padawan</b>	250 créditos	500 créditos
<b>Cavaleiro Jedi</b>	500 créditos	1000 créditos
<b>Mestre Jedi</b>	1000 créditos	1500 créditos
<b>Mestre do Conselho</b>	1500 créditos	2000 créditos

O valor da aposta é o requisito de entrada dos jogadores no duelo, desta forma, o duelo não pode ser criado se o jogador, autor do duelo, não possuir créditos suficientes para cobrir a aposta. Caso o jogador desafiado não tenha créditos suficientes para cobrir a aposta, o duelo assume o formato de desafio simples, com suas devidas regras, porém mantendo a recompensa do duelo. O sistema deverá recusar, no momento da criação do duelo, caso o nível de experiência do jogador desafiado não seja compatível com o nível de dificuldade do duelo.

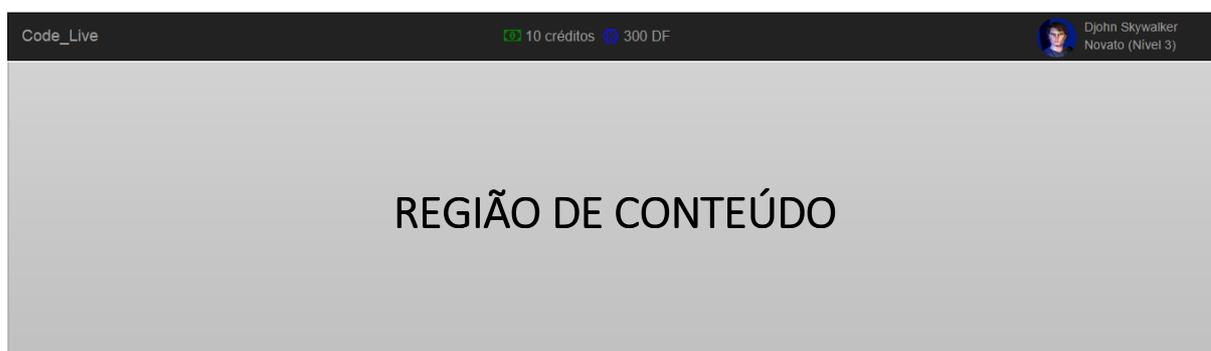
O vencedor do duelo recebe como recompensa 80% a mais sobre o valor da sua aposta e o dobro de experiência (DP) em relação ao ganho em um desafio comum (Tabela 8).

#### 4.5.4. Desenvolvimento do Protótipo

O foco inicial do desenvolvimento do software foi na interação e usabilidade básica do jogador com o ambiente. Deste modo, foi então planejada a construção da interface básica do ambiente utilizando recursos de web design como HTML, CSS e Javascript. Também foi utilizado o *framework* AngularJS que facilitou a criação das funcionalidades do *front-end* do ambiente, inclusive reduzindo a necessidade de *back-end*.

A interface do Code Live possui um layout por uma barra superior de informações e uma região de conteúdo. Essa barra superior contém informações importantes ao jogador e ficam sempre visíveis em qualquer tela do ambiente. Já a parte inferior é a destinada ao conteúdo das telas que varia de tela para tela. A figura a seguir contextualiza o layout do ambiente.

**Figura 26 – Layout da interface do Code Live**



Conforme a Figura 26, a barra superior contém informações importantes ao jogador, como o saldo atual de créditos e a quantidade de DF que o mesmo possui, ambas informações estão, respectivamente, presentes ao centro da barra. Já no canto direito da mesma barra estão presentes o nome de exibição do jogador, seu avatar e sua atual classe de experiência no Code Live, além de que, em seu nome há um link para acessar a tela de seu Perfil (seção 4.5.4.5).

As telas foram escolhidas a partir da sua importância e uso dentro do ambiente. Com isso, foram desenvolvidas cinco telas e três telas auxiliares:

1. Lista de Desafios
  - 1.1. Compra de Desafio
2. Modo de Criação
  - 2.1. Detalhes do Desafio
3. Modo Desafio
  - 3.1. Validação
4. Modo Duelo
  - 4.1. Validação
5. Perfil do Jogador

As telas auxiliares são janelas ou *modals* que sobrepõem as telas para apresentar informações e/ou proporcionar meios de interação com o usuário. A seguir, nas próximas seções, será apresentado detalhes de cada tela.

#### 4.5.4.1. Lista de Desafios

A tela Lista de Desafios é a tela inicial do ambiente na sua versão final, após a autenticação do usuário. Esta tela possui uma lista com todos os desafios publicados aos jogadores. Cada desafio é apresentado em uma caixa que contém informações importantes sobre o desafio. A figura a seguir ilustra a tela e suas características.

**Figura 27 – Tela Lista de Desafios**



A Figura 27 mostra a tela dividida em três áreas (1-laranja, 2-azul e 3-amarelo). A primeira área, à esquerda, seria o local destinado aos filtros de desafios que

permitiriam o jogador filtrar a lista dos desafios de acordo com suas preferências de níveis de dificuldade, classificação média, autor, etc. Já a área 2, ao centro, é composta por várias caixas, sendo que, cada uma delas representa um desafio diferente. A área 3 é composta por um botão que redireciona para a página de criação de novos desafios.

As cores predominantes nas caixas dos desafios representam o nível de dificuldade de cada desafio. A tabela a seguir ilustra a relação entre as cores e os níveis de dificuldade.

**Tabela 10 – Relação entre cores e níveis de dificuldade**

Cor	Nível de Dificuldade
Verde	Fundamentos da Força
Azul	Treinando com o Sabre de Luz
Laranja	Dominando a Força
Roxo	Controlando Mentes

A próxima figura mostra especificamente uma caixa de desafio, destacando as características importantes dos desafios presentes.

**Figura 28 – Caixa de desafio**



Conforme a Figura 28 mostra, existem diversas informações do desafio para o jogador, sendo elas: título do desafio (1), classificação média dos jogadores (2), autor do desafio (3), descrição resumida do desafio (4) e requisitos para compra do desafio (5). Ao clicar na caixa do desafio, caso o jogador tenha os requisitos mínimos, será apresentada uma *modal* com mais informações do desafio e possibilidades para jogá-lo, conforme ilustra a figura a seguir.

**Figura 29 – Tela de detalhes do desafio**



Conforme mostra a Figura 29, pode-se identificar vários elementos de um desafio dentro dessa janela de detalhes, como o título e o autor do desafio (1); média de nota dos jogadores e tags do desafio (2); tempo máximo de resolução do desafio (3); descrição detalhada do desafio (4); recompensas do desafio (5); botão para compra do desafio (6), para ser jogado como desafio simples; e o botão para compra como duelo (7).

Quando o jogador opta pela compra do desafio, uma confirmação de compra é exibida a ele, uma vez confirmado o desafio torna-se disponível para resolução. Já quando é escolhido pelo formato de duelo, uma nova tela é exibida (na mesma *modal*) solicitando ao jogador selecionar algumas informações do duelo, como: valor da aposta e seu oponente. A próxima figura ilustra a tela de compra de duelo.

**Figura 30 – Tela de compra de duelo**



O duelo permite que o jogador que está criando a partida possa personalizar algumas informações do duelo, como pode ser observado na Figura 30. Os valores limites da aposta (1) seguem as regras definidas no *Game Design* (Seção 4.5.3.4). O jogador deve também selecionar um oponente com quem deseja duelar (4) ou pode deixar o sistema escolher um oponente aleatório (3). O botão “Voltar para o desafio” (5) permite cancelar a compra como duelo e voltar para a tela anterior (Figura 29).

#### 4.5.4.2. Modo Criação

A tela do Modo Criação, conforme descreve o game design (seção 4.5.3.2), é utilizada na criação de novos desafios que poderão posteriormente serem resolvidos por outros jogadores na forma de desafio comum ou duelo. A figura a seguir é uma captura da tela.

**Figura 31 – Tela de criação de desafios**

Como pode ser observado na Figura 31, a tela de cadastro de desafios é composta por diversos campos, sendo alguns de preenchimento obrigatório. Existe também uma área composta por três botões que efetuam ações sobre o desafio que está sendo criado. Os itens do formulário são descritos a seguir.

**Título do Desafio:** Título que identificará o desafio para os demais jogadores após ser publicado.

**Tags:** Palavras ou termos utilizados para categorizar os desafios. É utilizado também na filtragem de desafios.

**Tempo Máximo para Resolução:** Período máximo de tempo que o jogador terá para solucionar o desafio. Esse período pode ser em horas ou dias.

**Entradas:** Vetor de valores que será utilizado pelo sistema para validar o código criado pelo jogador. Cada registro desse vetor é composto por nome de variável e um valor. O jogador deverá utilizar o método “leiaEntrada(nomeVariavel)” para receber essas entradas em seu código. O jogador não tem acesso aos valores mencionados.

**Saídas:** Vetor de valores que deverão ser a saída da impressão do código do jogador. A ordem em que as saídas são inseridas, que pode ser alterada posteriormente, é relevante, já que o sistema avaliará as impressões do usuário usando essa mesma ordem.

**Nível de Dificuldade:** Nível de dificuldade do desafio que está sendo criado. O criador deve escolher o nível de acordo com o enunciado que será inserido no desafio. Ao clicar em cada nível são apresentadas informações sobre o mesmo, como os

requisitos que o enunciado precisa ter, conforme a Tabela 5, presente no Game Design (seção 4.5.3.2). Alguns níveis de dificuldade não são acessíveis a todos, conforme a Tabela 6, por isso, os níveis que o jogador não tiver acesso ficarão inativos nessa tela.

**Descrição Resumida:** Descrição simples do desafio. Essa descrição será apresentada ao jogador, portanto deve-se evitar dar muitos detalhes sobre o enunciado para não favorecer o jogador antes que o mesmo inicie o desafio.

**Enunciado:** É a descrição do problema proposto para resolução por meio do desafio. É imprescindível que o enunciado tenha uma linguagem o mais entendível possível, já que o jogador depende exclusivamente deste texto para entender o que ele precisa criar para solucionar o desafio. É preferível que o enunciado seja construído usando uma linguagem baseada na temática do ambiente, o universo *Star Wars*.

**Pré-código:** É um código que será previamente carregado na tela de resolução do desafio.



**Botão Salvar:** Botão utilizado para salvar o desafio na conta do jogador para edição posterior. Este botão não deixa o desafio disponível para outros jogadores.



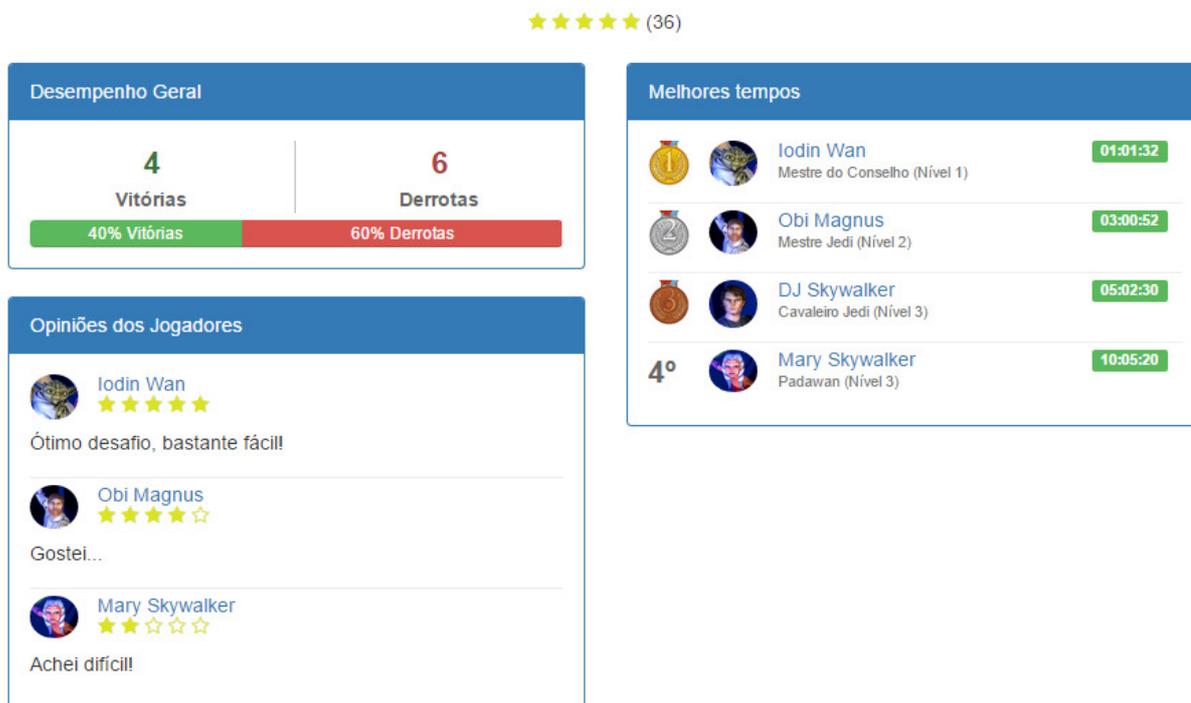
**Botão Publicar:** Utilizado para publicar um desafio para os demais jogadores. Uma vez publicado o desafio não pode ser alterado, apenas desativado. A publicação gera custos para o criador de acordo com o nível de dificuldade do desafio, conforme a Tabela 6 do Game Design.



**Botão Visualizar:** Abre uma modal idêntica a utilizada na tela de desafios (Figura 29) com os atuais dados inseridos no formulário de cadastro de desafio, criando assim uma simulação de como ficará o desafio ao ser publicado.

Uma vez que o desafio é criado, o jogador autor do mesmo pode acessar as estatísticas, opiniões e as respostas dos jogadores. Para tanto, o jogador precisa acessar o seu desafio pela tela de desafios (seção 4.5.4.1). A *modal* com os detalhes do desafio é diferente da anteriormente apresentada (Figura 27), já que o autor não pode jogar o desafio. A próxima figura ilustra a tela de detalhes do desafio na visão de seu autor.

**Figura 32 – Modal de detalhes do desafio na visão de seu autor**



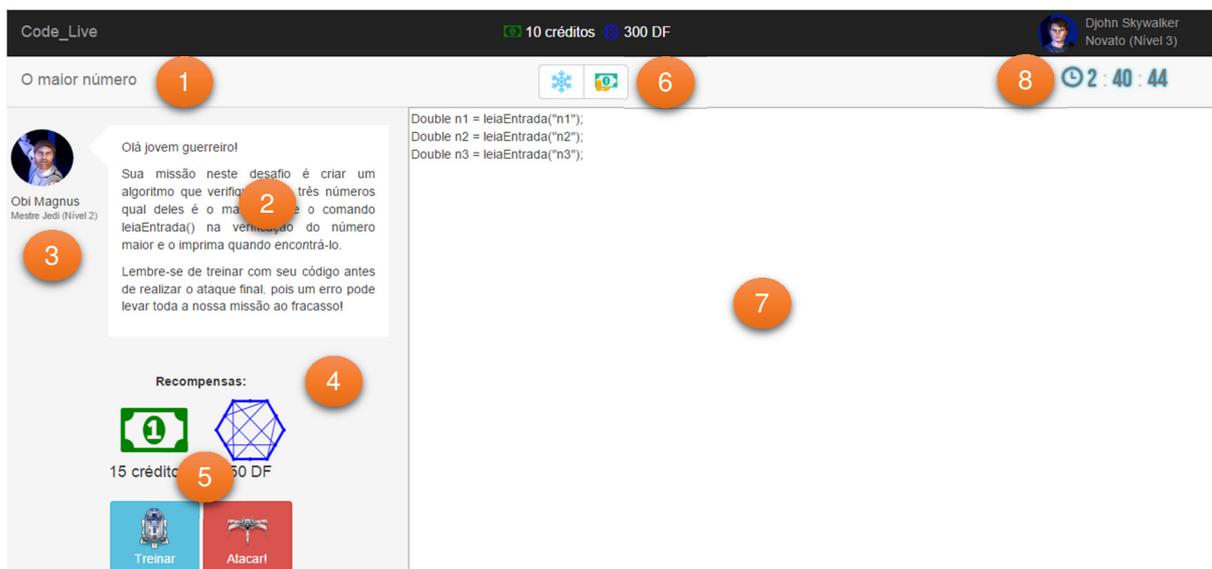
Conforme a Figura 32, a tela de detalhes do desafio traz as estatísticas e opiniões dos jogadores que participaram do desafio em questão. Na primeira caixa, à esquerda, é apresentado o desempenho geral de todos os jogadores participantes, com o número e percentual de vitórias e derrotas. A segunda caixa, à direita, traz o ranking dos melhores tempos dos jogadores que conseguirem concluir com vitória o desafio. E a última caixa contém as opiniões dos jogadores, incluindo a classificação em estrelas dada por cada um.

É possível o autor acessar também os códigos de cada jogador que respondeu o desafio, para isso basta clicar em nos links presentes nos nomes dos jogadores dentro da tela de detalhes do desafio.

#### 4.5.4.3. Modo Desafio

A tela Modo Desafio compreende a parte do ambiente onde o jogador irá resolver seus desafios adquiridos. Esta tela também possui informações básicas sobre o desafio, além de fornecer recursos de auxílio para resolução do desafio. A figura a seguir ilustra a tela em questão.

**Figura 33 – Tela de resolução do desafio (Modo Desafio)**



Conforme a Figura 33, a tela do Modo Desafio apresenta algumas informações básicas do desafio, como o título (1), enunciado (2), autor do desafio (3), recompensas (4) e tempo restante para resolução (8), que é representado por um contador regressivo. Grande parte da tela é preenchida pela área dedicada a inserção do código da resolução do desafio (7). Como recursos de auxílio existem os poderes (6) e os botões “Treinar” e “Atacar!” (5).

Os Poderes são recursos ou auxílios que podem ser utilizados durante desafios ou duelos para ajudar o jogador na sua resolução. Os poderes presentes nesta tela são, respectivamente, “Congelar o tempo” e “O dobro ou nada”, que seguem as mesmas características definidas na Tabela 4 do *Game Design*.

O botão “Treinar” permite o jogador executar o seu código de forma que possa testá-lo em busca de erros. A execução funciona semelhantemente à do sistema JLive 2 (seção 3.1.4), onde o código é executado e um log com as impressões feitas pelo jogador ou erros na compilação são apresentados.

Já o botão “Atacar!” deve ser utilizado pelo jogador quando este tiver a certeza que o código está correto, pois uma vez acionado o botão não tem volta, caso o código esteja errado, o jogador é dado como derrotado. Para identificar se houve erro ou não no código, o Code Live compara as saídas do jogador com as saídas pré-definidas pelo criador do desafio, se forem compatíveis, o usuário vence. Uma *modal* é apresentada durante e depois, com os resultados, conforme a figura a seguir.

**Figura 34 – Modal de compilação, validação e resultados do desafio**



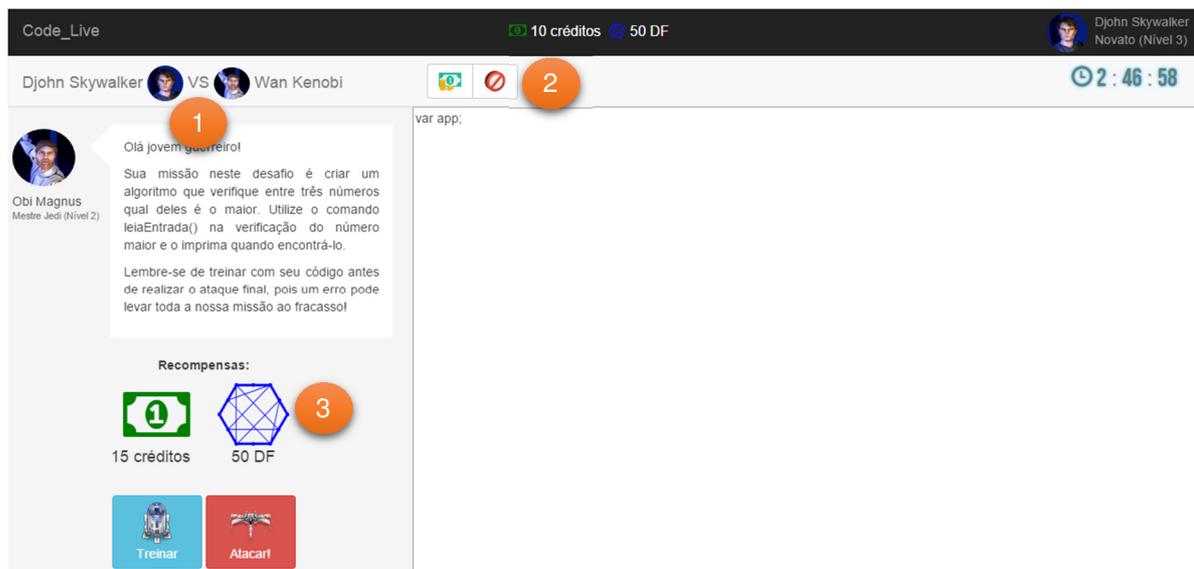
Como pode ser percebido na Figura 34, essa modal em questão possui duas telas, cada uma é apresentada em momentos diferentes do processo. A tela 1 é exibida durante o processo de compilação e a tela 2 possui os resultados da compilação, podendo ser uma vitória ou derrota.

A tela 1 também possui uma animação com um dos personagens do universo SW, o robô R2-D2. Já na tela 2 são apresentados os resultados da compilação, neste caso, o jogador obteve vitória. A tela 2, seja quando há derrota ou vitória, apresentará o recurso de avaliação do desafio, que fica na parte inferior da tela, como também os botões “Minhas Estatísticas” que leve o jogador à tela de Perfil (seção 4.5.4.5), e o “Lista de Desafios” que leva a tela de mesmo nome (seção 4.5.4.1). Quando o jogador obtém vitória é apresentado as recompensas que conquistou no desafio.

#### **4.5.4.4. Modo Duelo**

A tela do Modo Duelo é muito semelhante a tela do Modo Desafio, assim como suas características. Um duelo é um desafio entre dois jogadores, seu diferencial em relação ao Modo Desafio é que vence o jogador que resolver o desafio corretamente em menos tempo. Além disso as recompensas neste modo também diferem. Todas as particularidades e detalhes do funcionamento deste modo estão descritas no *Game Design* (seção 4.5.3.4). A figura a seguir ilustra a tela em questão.

**Figura 35 – Tela de resolução de duelo (Modo Duelo)**



Em primeiro instante pode-se perceber a grande semelhança entre a tela apresentada na figura anterior (Figura 35) e a tela do Modo Desafio (Figura 33), inclusive a maioria dos recursos e botões presentes funcionam da mesma maneira em ambas as telas. Porém, existem três diferenças entre as duas telas.

A primeira diferença entre as duas telas encontra-se na parte superior esquerda da tela (Figura 35.1), que era dedicada ao título do desafio, já nesta tela o título oscila em um intervalo de tempo com os nomes dos dois jogadores que estão duelando.

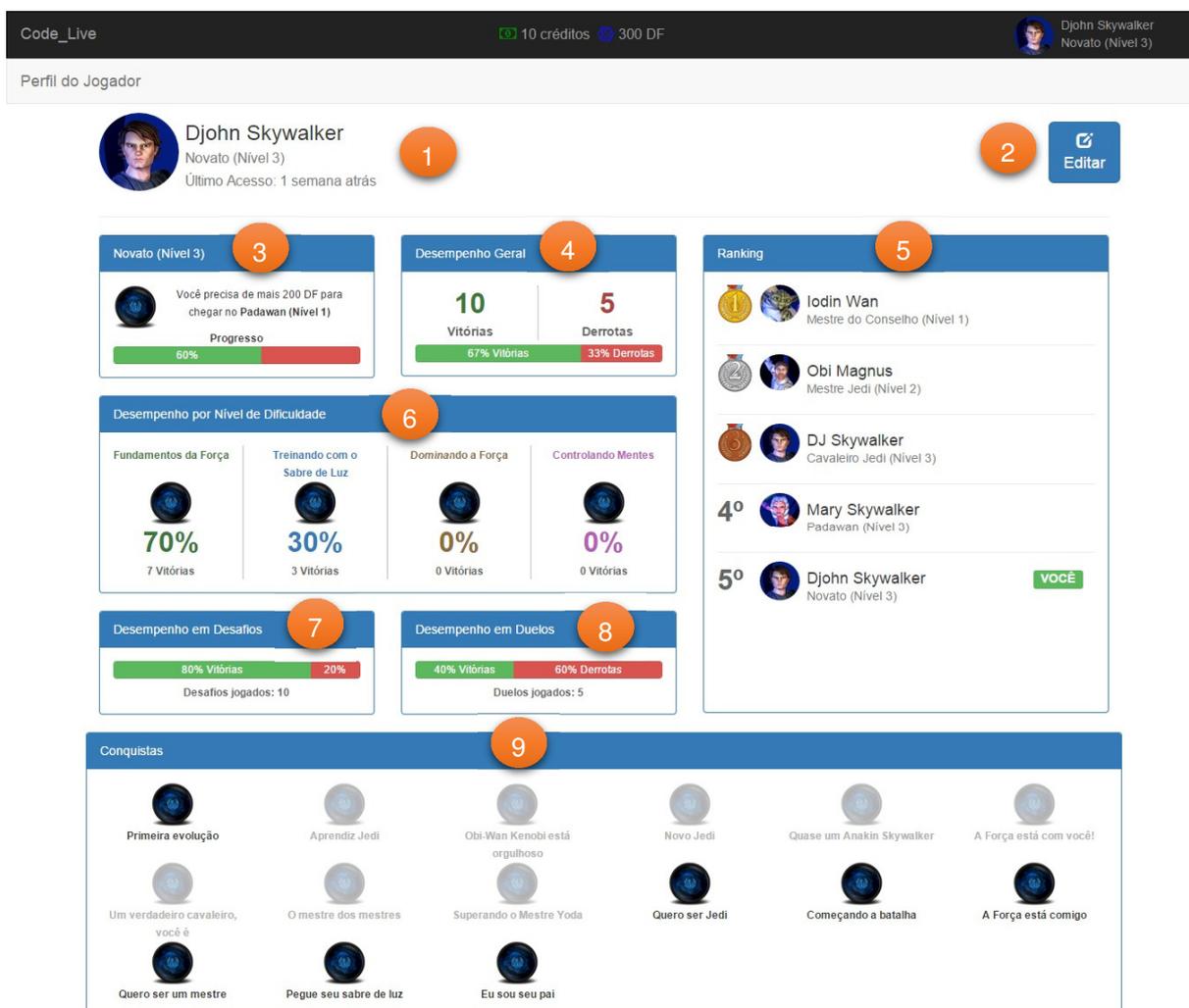
A segunda diferença está nos poderes disponíveis, conforme determina a Tabela 4 do *Game Design*, alguns poderes só podem ser utilizados em desafios normais, assim como, outros só podem ser utilizados em duelos.

A terceira e última diferença condiz com as recompensas que, como mencionado anteriormente, são diferenciadas em relação à desafios de mesmo nível de dificuldade no Modo Desafio.

#### **4.5.4.5. Perfil do Jogador**

A tela de Perfil do Jogador é onde são apresentadas diversas informações estatísticas do jogador, como desempenho (número de vitórias e derrotas), posição no ranking, entre outros. A figura a seguir apresenta a tela em questão.

Figura 36 – Tela de Perfil do Jogador



Conforme mostra a Figura 36, a tela do Perfil do Jogador é bastante extensa e contém muitas informações sobre o jogador, incluindo gráficos e estatísticas percentuais. Os elementos existentes na tela são, conforme a numeração da figura:

1. Nome, classe e nível de experiência e data do último acesso do jogador;
2. Botão de edição dos dados do jogador;
3. Progresso de nível de experiência do jogador;
4. Desempenho geral do jogador;
5. Ranking de todos os jogadores;
6. Desempenho por nível de dificuldade do desafio;
7. Desempenho do jogador em desafios simples;
8. Desempenho do jogador em duelos; e
9. Conquistas do jogador.

O item 3 além de trazer o percentual de progresso no nível atual do jogador, também traz a informação da quantidade de DF necessário para atingir o próximo nível de experiência.

O item 4 avalia o quantitativo de vitórias e derrotas do jogador, trazendo um comparativo percentual entre os dois estados. Já os itens 6, 7 e 8 utilizam as vitórias e derrotas de forma mais específica, como o item 6 que avalia o quantitativo de vitórias por nível de dificuldade, sem levar em consideração as derrotas nessa análise. Em contrapartida, o item 7, semelhantemente ao item 4, avalia as vitórias e derrotas, mas somente no âmbito dos desafios simples, quanto o item 8 faz o mesmo, porém avaliando os duelos.

Um item 5 é um ranking ou quadro de líderes de todos os jogadores ativos do Code Live. Esse ranking é baseado na quantidade de DF que cada jogador possui, quanto maior o DF, mais alta a posição no ranking. Um jogador verá seu nome em destaque no ranking através do rótulo “VOCÊ”. O Code Live sempre dará prioridade a visualização da posição do jogador em questão, que poderá utilizar a barra de rolagem para visualizar os demais jogadores, caso esta esteja disponível.

Por fim, o item 9 é um mural de conquistas do jogador. Conforme foi determinado na Tabela 3 do Game Design, existem 15 conquistas disponíveis a todos os jogadores. As conquistas apresentadas de forma transparente são as que ainda não foram obtidas pelo jogador.

Estas foram as telas planejadas para compor o protótipo, tendo em vista que estas possuem as funcionalidades mais importantes do Code Live. Vale ressaltar que todas as telas possuem interatividade básica, ou seja, funcionam como o ambiente real, porém sem o processamento e armazenamento no banco de dados.

## 5 CONSIDERAÇÕES FINAIS

Como apresentado ao longo deste trabalho, foi mostrado que mesmo com as diversas ferramentas de suporte à programação como as IDE's, ainda existe a necessidade de soluções que auxiliem e incentivem de alguma forma o aprendizado e a prática de programação.

Com isso, foi proposto um processo de gamificação sobre um ambiente virtual de programação chamado JLive 2, surgindo assim o projeto Code Live. O projeto utilizou vários elementos de gamificação juntamente com elementos do universo da saga Star Wars, como uma forma de obter uma plataforma divertida e de crescimento contínuo, já que a mesma permite que os jogadores acrescentem novos conteúdos e a gamificação os incentiva a isso.

Dentre os perfis de jogadores esperados, um tipo que pode mais se beneficiar com o ambiente são os alunos dos cursos de computação, principalmente os iniciantes. O Code Live usa em seus desafios uma linguagem de programação simplificada, perfeita para quem está começando a aprender programar. Apesar de não ser seu objetivo principal, o Code Live tem grande potencial para gerar conhecimento e aprendizado entre esses jogadores.

Os outros tipos de jogadores, como os programadores mais experientes, tornam-se importante ao ambiente, pois serão eles os principais criadores de conteúdo. Quanto maior a diversidade de conteúdo, maiores são as chances dos jogadores se manterem jogando, garantido assim a diversão por mais tempo.

O protótipo criado e apresentado neste trabalho traz somente os pontos críticos do Code Live. O ambiente, em seu planejamento inicial, conta com muitos outros recursos, como: manual de instruções, gerenciamento de turmas/equipes, dentre outros. O protótipo também se limitou a metade do universo SW, os Jedi, ficando para posterioridades a implantação do universo Sith.

### 5.1. Trabalhos Futuros

Como mencionado anteriormente, o protótipo não contemplou todas as funcionalidades planejadas para o Code Live, ficando assim para trabalhos futuros a

tarefa de implementar e implantar, de fato, o Code Live. A lista a seguir apresenta alguns recursos e funcionalidades previstas para serem acrescentadas ao ambiente:

- Elaborar o manual de instruções, incluindo explicações do funcionamento da linguagem de programação utilizada;
- Melhorias na gamificação, adicionando mais elementos ao contexto atual;
- Universo Sith (Star Wars), incluindo a opção de o jogador escolher entre as duas facções no momento do seu cadastro;
- Mais integração social (redes sociais, amigos etc);
- Melhorar a validação dos códigos, deixando de ser apenas uma variável binária e passando a identificar um percentual de acerto; e
- Gerar aprendizado real.

Dentre as melhorias propostas, as duas últimas têm um nível de complexidade muito grande de tal forma que nem entraram no planejamento inicial do Code Live. O sistema de validação, como proposto acima, necessitaria de uma avaliação inteligente do código, algo que, inclusive, é até tema de dissertações e outros trabalhos. A parte da geração de aprendizagem exigiria montar uma equipe de especialistas na área além de um grupo de possíveis usuários alvos dessa aprendizagem.

## 6 REFERÊNCIAS BIBLIOGRÁFICAS

ARAÚJO, Karina. OS JOGOS E A EDUCAÇÃO. **Revista Eletrônica de Educação** , 2011.

BATISTA, Mônica de Lourdes *et al.* UM ESTUDO SOBRE A HISTÓRIA DOS JOGOS ELETRÔNICOS. **Revista Eletrônica da Faculdade Metodista Granbery** , 2007. Disponível em: <<http://re.granbery.edu.br/artigos/MjQ4.pdf>>. Acesso em: 6 nov. 2014.

CAMPBELL-KELLY, Martin. *The Edsac Simulator*. Disponível em: <<http://www.dcs.warwick.ac.uk/~edsac/>>.

CANALTECH. *Code Hunt ajuda você a aprender programação brincando*. Disponível em: <<http://canaltech.com.br/dica/jogos-online/Code-Hunt-ajuda-voce-a-aprender-programacao-brincando>>. Acesso em: 18 nov. 2014.

CARDOSO. Djonathas; GOMES. Jackson. **JLIVE 2**. Palmas-TO: [s.n.], 2014.

COFFEY, HEATHER. *Digital game-based learning*. Disponível em: <<http://www.learnnc.org/lp/pages/4970>>. Acesso em: 8 nov. 2014.

CRAWFORD, Chris; PEABODY, Sue; LOPER, Donna. *The Art of Computer Game Design* by Chris Crawford. p. 81 , 2003.

DANVANÇO, Bia. *Ludificação ou Gamificação?* Disponível em: <<http://opusphere.com/ludificacao-ou-gamificacao/>>. Acesso em: 10 dez. 2014.

DETERDING, Sebastian *et al.* From Game Design Elements to Gamefulness : Defining “ Gamification ”. p. 9–11 , 2011.

DEUBEL, Patricia. Game on! Now Educators Can Translate Their Students’ Love of Video Games into the Use of a Valuable, Multifaceted Learning Tool. **T H E Journal (Technological Horizons In Education)** , 2006.

EDMAN, Lenore. *Play Tennis for Two in Florida and Arizona*. Disponível em: <<http://www.evilmadscientist.com/2011/play-tennis-for-two-in-florida-and-arizona/>>. Acesso em: 10 dez. 2014.

EPPER, Rhonda M.; DERRYBERRY, Anne; JACKSON, Sean. *Game-Based Learning*. p. 1–11 , 2012.

FARDO, Marcelo. A GAMIFICAÇÃO APLICADA EM AMBIENTES DE APRENDIZAGEM. **CINTED-UFRGS** p. 1–9 , 2013.

FITZ-WALTER, Zac. *A brief history of gamification*. Disponível em: <<http://zefcan.com/2013/01/a-brief-history-of-gamification/>>.

GOMES, Anabela; HENRIQUES, Joana; MENDES, António José. Uma proposta para ajudar alunos com dificuldades na aprendizagem inicial de programação de computadores. p. 93–103 , 2008.

GONÇALVES, Janaina; NUNES, José Mauro. JOGOS ELETRÔNICOS E SUBJETIVIDADE. PONTOS PARA UMA DISCUSSÃO ACERCA DO FUNCIONAMENTO COGNITIVO. , 2010.

GOOGLE. *AngularJS — Superheroic JavaScript MVW Framework*. Disponível em: <<https://angularjs.org/>>. Acesso em: 17 jun. 2015.

GOOGLE. *Google Trends*. Disponível em: <<http://www.google.com/trends/explore#q=gamification>>. Acesso em: 10 dez. 2014.

GUIMARÃES. Thiago et al. **BRAZIL SOCIAL MEDIA: The Mobile Middle Class Goes Social**. [S.l: s.n.], 2013.

HOUAISS, Antonio. *Dicionário Eletrônico Houaiss da Língua Portuguesa* . Rio de Janeiro: Objetiva. , 2001

KAPP, Karl. **The gamification of learning and instruction: game-based methods and strategies for training and education**. São Francisco: Pfeiffer, 2012. 201-206 p.

KUUTTI, Julius. Designing Gamification. **Marketing,(May)** n. May , 2013. Disponível em: <<http://herkules.oulu.fi/thesis/nbnfioulu-201305311405.pdf>>. Acesso em: 14 out. 2014.

LEE, Joey; HAMMER, Jessica. Gamification in Education : What , How , Why Bother ? What : Definitions and Uses. **Academic Exchange Quarterly** v. 15, p. 1–5 , 2011.

LUCASFILM. *Star Wars: The Clone Wars* . [S.l: s.n.]. , 2008

MARCZEWSKI, Andrzej. Gamification: A Simple Introduction. p. 153 , 2013.

MOSS, Richard. *Let Me Edutain You: Buy My Lemonade!* Disponível em: <<http://archive.vg/blog/10-feature/let-me-edutain-you/25-let-me-edutain-you-buy-my-lemonade>>. Acesso em: 15 nov. 2014.

OXO. **Wikipédia: a enciclopédia livre**. Flórida: Wikimedia Foundation, 2014. Disponível em: <<https://en.wikipedia.org/w/index.php?title=OXO&oldid=618817998>>. Acesso em: 20 nov. 2014.

PETRÓ, Gustavo. *Dificuldade excessiva faz game para smartphones “Flappy Bird” virar hit*. Disponível em: <<http://g1.globo.com/tecnologia/games/noticia/2014/02/dificuldade-excessiva-faz-game-para-smartphones-flappy-bird- virar-hit.html>>. Acesso em: 16 nov. 2014.

PRADO, Maria Elisabette. LOGO - LINGUAGEM DE PROGRAMAÇÃO E AS IMPLICAÇÕES PEDAGÓGICAS. **Nied-Unicamp** , 2000.

PRENSKY, Marc. Digital Natives, Digital Immigrants Part 1. **On the Horizon** v. 9, n. 5, p. 1–6 , 2001.

SALEN, Katie; ZIMMERMAN, Eric. **Rules of Play - Game Design Fundamentals**. [S.l.]: Massachusetts Institute of Technology, 2004. .

SOARES, Alexandre. *ProjetoLogo - Uma Apresentação*. Disponível em: <<http://projetologo.webs.com/texto1.html>>. Acesso em: 18 nov. 2014.

SOARES, Alexandre. *ProjetoLogo - xLogo*. Disponível em: <<http://projetologo.webs.com/xlogo.html>>. Acesso em: 18 nov. 2014.

STAR WARS WIKI. *A Força*. Disponível em: <<http://pt.starwars.wikia.com/wiki/For%C3%A7a>>. Acesso em: 21 jun. 2015a.

STAR WARS WIKI. *Ordem Jedi*. Disponível em: <[http://pt.starwars.wikia.com/wiki/Ordem\\_Jedi](http://pt.starwars.wikia.com/wiki/Ordem_Jedi)>. Acesso em: 21 jun. 2015b.

STAR WARS WIKI. *Star Wars*. Disponível em: <[http://pt.starwars.wikia.com/wiki/Star\\_Wars](http://pt.starwars.wikia.com/wiki/Star_Wars)>. Acesso em: 21 jun. 2015c.

STAR WARS WIKI. *Star Wars Wiki em Português*. Disponível em: <[http://pt.starwars.wikia.com/wiki/P%C3%A1gina\\_principal](http://pt.starwars.wikia.com/wiki/P%C3%A1gina_principal)>. Acesso em: 21 jun. 2015d.

TAROUCO, Liane Margarida *et al.* Jogos educacionais. **CINTED** p. 1–7 , 2004. Disponível em: <[http://www.virtual.ufc.br/cursouca/modulo\\_3/Jogos\\_Educacionais.pdf](http://www.virtual.ufc.br/cursouca/modulo_3/Jogos_Educacionais.pdf)>. Acesso em: 5 nov. 2014.

WERBACH, Kevin; HUNTER, Dan. **For the win: How game thinking can revolutionize your business**. [S.l: s.n.], 2012. .9781613630228.

WIKIA. *About*. Disponível em: <[http://www.wikia.com/About\\_Us](http://www.wikia.com/About_Us)>. Acesso em: 21 jun. 2015.

WINTER, David. *Noughts And Crosses - The oldest graphical computer game*. Disponível em: <<http://www.pong-story.com/1952.htm>>.

WOOKIEEPEDIA. *Wookieepedia, the Star Wars Wiki*. Disponível em: <[http://starwars.wikia.com/wiki/Main\\_Page](http://starwars.wikia.com/wiki/Main_Page)>. Acesso em: 21 jun. 2015.

ZICBERMANN, G; CUNNINGHAM, C. **Gamification by Design**. [S.l.]: O'REILLY, 2011. .9781449397678.