



Universidade de Brasília – UnB  
Faculdade UnB Gama – FGA  
Engenharia de Software

**Gamificação aplicada à educação: Ferramenta  
de apoio ao ensino e aprendizagem de  
programação.**

Autor: Eduardo Júnio Veloso Rodrigues  
Orientador: Dr. Wander Cleber Maria Pereira da Silva

Brasília, DF  
2020



Eduardo Júnio Veloso Rodrigues

**Gamificação aplicada à educação: Ferramenta de apoio  
ao ensino e aprendizagem de programação.**

Monografia submetida ao curso de graduação  
em Engenharia de Software da Universidade  
de Brasília, como requisito parcial para ob-  
tenção do Título de Bacharel em Engenharia  
de Software.

Universidade de Brasília – UnB

Faculdade UnB Gama – FGA

Orientador: Dr. Wander Cleber Maria Pereira da Silva

Brasília, DF

2020

---

Eduardo Júnio Veloso Rodrigues

Gamificação aplicada à educação: Ferramenta de apoio ao ensino e aprendizagem de programação./ Eduardo Júnio Veloso Rodrigues. – Brasília, DF, 2020-  
50 p. : il. (algumas color.) ; 30 cm.

Orientador: Dr. Wander Cleber Maria Pereira da Silva

Trabalho de Conclusão de Curso – Universidade de Brasília – UnB  
Faculdade UnB Gama – FGA , 2020.

1. Palavra-chave01. 2. Palavra-chave02. I. Dr. Wander Cleber Maria Pereira da Silva. II. Universidade de Brasília. III. Faculdade UnB Gama. IV. Gamificação aplicada à educação: Ferramenta de apoio ao ensino e aprendizagem de programação.

CDU 02:141:005.6

---

Eduardo Júnio Veloso Rodrigues

## **Gamificação aplicada à educação: Ferramenta de apoio ao ensino e aprendizagem de programação.**

Monografia submetida ao curso de graduação em Engenharia de Software da Universidade de Brasília, como requisito parcial para obtenção do Título de Bacharel em Engenharia de Software.

Trabalho aprovado. Brasília, DF, 03 de Julho de 2020 – Data da aprovação do trabalho:

---

**Dr. Wander Cleber Maria Pereira da  
Silva**  
Orientador

---

**Titulação e Nome do Professor**  
**Convidado 01**  
Convidado 1

---

**Titulação e Nome do Professor**  
**Convidado 02**  
Convidado 2

Brasília, DF  
2020

**A dedicatória é opcional. Caso não deseje uma, deixar todo este arquivo em  
branco.**

*Este trabalho é dedicado às crianças adultas que,  
quando pequenas, sonharam em se tornar cientistas.*

# Agradecimentos

A inclusão desta seção de agradecimentos é opcional, portanto, sua inclusão fica a critério do(s) autor(es), que caso deseje(em) fazê-lo deverá(ão) utilizar este espaço, seguindo a formatação de *espaço simples e fonte padrão do texto (sem negritos, aspas ou itálico)*.

**Caso não deseje utilizar os agradecimentos, deixar toda este arquivo em branco.**

A epígrafe é opcional. Caso não deseje uma, deixe todo este arquivo em  
branco.

*“Não vos amoldeis às estruturas deste mundo,  
mas transformai-vos pela renovação da mente,  
a fim de distinguir qual é a vontade de Deus:  
o que é bom, o que Lhe é agradável, o que é perfeito.  
(Bíblia Sagrada, Romanos 12, 2)*

# Resumo

O resumo deve ressaltar o objetivo, o método, os resultados e as conclusões do documento. A ordem e a extensão destes itens dependem do tipo de resumo (informativo ou indicativo) e do tratamento que cada item recebe no documento original. O resumo deve ser precedido da referência do documento, com exceção do resumo inserido no próprio documento. (...) As palavras-chave devem figurar logo abaixo do resumo, antecedidas da expressão Palavras-chave:, separadas entre si por ponto e finalizadas também por ponto. O texto pode conter no mínimo 150 e no máximo 500 palavras, é aconselhável que sejam utilizadas 200 palavras. E não se separa o texto do resumo em parágrafos.

**Palavras-chave:** latex. abntex. editoração de texto.



# Abstract

This is the english abstract.

**Key-words:** latex. abntex. text editoration.

# Lista de ilustrações

Figura 1 – Sequência de passos típicos na apresentação de uma disciplina . . . . .	17
Figura 2 – Divisão do octógono . . . . .	24
Figura 3 – Gamificação <i>White hat</i> e <i>Black hat</i> . . . . .	25
Figura 4 – As fases da jornada de um jogador. . . . .	26
Figura 5 – Perfil de jogadores . . . . .	29
Figura 6 – <i>Core drives</i> mais utilizados de acordo com as fases e perfis de jogadores. . . . .	29
Figura 7 – Exemplo de DER . . . . .	31
Figura 8 – Personagem Aisha . . . . .	34
Figura 9 – Personagem Voxter . . . . .	35
Figura 10 – Personagem Lince . . . . .	36
Figura 11 – Personagem Amazona . . . . .	37
Figura 12 – Personagem Tymer . . . . .	38
Figura 13 – Personagem Scar . . . . .	39
Figura 14 – DER da ferramenta . . . . .	41
Figura 15 – DL da ferramenta . . . . .	42

# Lista de tabelas

Tabela 1	–	Principais <i>core drives</i> envolvidos em cada perfil de jogador . . . . .	28
Tabela 2	–	Requisitos implementados . . . . .	32
Tabela 3	–	Conteúdos por <i>level</i> . . . . .	40

# Sumário

<b>1</b>	<b>INTRODUÇÃO</b>	<b>13</b>
<b>1.1</b>	<b>Contexto</b>	<b>13</b>
<b>1.2</b>	<b>Justificativa</b>	<b>15</b>
<b>1.3</b>	<b>Problema</b>	<b>16</b>
<b>1.4</b>	<b>Objetivos</b>	<b>16</b>
<b>2</b>	<b>REFERENCIAL TEÓRICO</b>	<b>17</b>
<b>2.1</b>	<b>Aprendizagem de programação</b>	<b>17</b>
<b>2.2</b>	<b>Gamificação</b>	<b>17</b>
2.2.1	Gamificação na Educação	18
<b>2.3</b>	<b><i>Framework Octalysis</i></b>	<b>19</b>
2.3.1	Significado épico e chamado	19
2.3.2	Desenvolvimento e Conquista	20
2.3.3	Empoderamento e feedback	21
2.3.4	Propriedade e posse	21
2.3.5	Influência e relacionamento social	22
2.3.6	Escassez e impaciência	22
2.3.7	Imprevisibilidade e curiosidade	23
2.3.8	Prevenção e perda	23
<b>2.4</b>	<b>Elementos motivacionais: Octógono</b>	<b>24</b>
2.4.1	Divisão dos motivadores de acordo com sua natureza	24
2.4.2	Gamificação <i>White Hat</i> e <i>Black Hat</i>	24
2.4.3	Níveis <i>Octalysis</i>	25
2.4.3.1	Nível 1	26
2.4.3.2	Nível 2	26
2.4.3.3	Nível 3	27
2.4.3.3.1	<i>Design</i> de gamificação: <i>Octalysis</i> e <i>Bartle</i>	27
2.4.3.4	Perfis de Jogadores	27
2.4.3.4.1	Predadores ( <i>Killers</i> )	27
2.4.3.4.2	Realizadores ( <i>Achievers</i> )	27
2.4.3.4.3	Exploradores ( <i>Explorers</i> )	28
2.4.3.4.4	Socializadores ( <i>Socializers</i> )	28
2.4.3.5	Relação entre <i>Core drives</i> e Perfis de jogadores	28
<b>2.5</b>	<b>Modelagem de banco de dados</b>	<b>30</b>
2.5.1	Diagrama entidade-relacionamento	30

2.5.1.1	Entidade . . . . .	30
2.5.1.2	Relacionamento . . . . .	30
2.5.2	Diagrama Lógico . . . . .	31
<b>2.6</b>	<b><i>Core drives</i></b> . . . . .	<b>31</b>
<b>3</b>	<b>METODOLOGIA</b> . . . . .	<b>32</b>
<b>3.1</b>	<b>Requisitos</b> . . . . .	<b>32</b>
<b>4</b>	<b>CARACTERÍSTICAS DA FERRAMENTA</b> . . . . .	<b>33</b>
<b>4.1</b>	<b>Narrativa</b> . . . . .	<b>33</b>
<b>4.2</b>	<b>Personagens</b> . . . . .	<b>33</b>
4.2.1	Aisha . . . . .	33
4.2.2	Voxter . . . . .	35
4.2.3	Lince . . . . .	36
4.2.4	Amazona . . . . .	37
4.2.5	Tymer . . . . .	38
4.2.6	Scar . . . . .	39
<b>4.3</b>	<b>Questões</b> . . . . .	<b>39</b>
<b>4.4</b>	<b><i>Levels</i></b> . . . . .	<b>40</b>
<b>4.5</b>	<b>Desafios</b> . . . . .	<b>40</b>
4.5.1	Jogador contra Jogador . . . . .	40
4.5.2	Desafio individual . . . . .	40
<b>5</b>	<b>SOLUÇÃO TÉCNICA</b> . . . . .	<b>41</b>
<b>5.1</b>	<b>Modelagem de dados</b> . . . . .	<b>41</b>
	<b>REFERÊNCIAS</b> . . . . .	<b>43</b>
	<b>APÊNDICES</b> . . . . .	<b>45</b>
	<b>APÊNDICE A – PRIMEIRO APÊNDICE</b> . . . . .	<b>46</b>
	<b>APÊNDICE B – SEGUNDO APÊNDICE</b> . . . . .	<b>47</b>
	<b>ANEXOS</b> . . . . .	<b>48</b>
	<b>ANEXO A – PRIMEIRO ANEXO</b> . . . . .	<b>49</b>
	<b>ANEXO B – SEGUNDO ANEXO</b> . . . . .	<b>50</b>

# 1 Introdução

## 1.1 Contexto

A tecnologia tem vindo a evoluir rapidamente e, com isto, nota-se o surgimento de diversos desafios. Um destes desafios é a falta de programadores que possuam competências e qualificação necessárias para solucionar os mais diversos problemas presentes nos mais variados projetos. Este fato está relacionado com a metodologia adotada no ensino de programação e também com a falta de motivação por parte dos estudantes (PEREIRA; COSTA; APARICIO, 2017). Dai, Zhao e Chen (2010) afirmam que, uma das razões de os alunos não absorverem eficientemente os conceitos relacionados à programação se dá pela falta de concentração e motivação dos mesmos frente a exposição destes conteúdos na forma tradicional.

Thais et al. (2002) destaca que a aprendizagem dos conceitos e mecanismos envolvidos na construção de programas não é trivial, uma vez que requer a utilização de raciocínio na sua forma mais abstrata. Um dos problemas mais comuns segundo os autores são: dificuldades no entendimento de comandos, sintaxe dos comandos, dificuldades em entender os resultados da execução de um determinado comando pela máquina, dificuldades em dar os primeiros passos relativos ao estudo de programação entre outros.

Almeida et al. (2019) diz que, em geral, os alunos têm grandes dificuldades em compreender e aplicar os conceitos relativos à programação. Uma das grandes dificuldades está relacionada a problemas de compreensão e aplicação de noções básicas, como por exemplo o uso de estruturas de controle e estruturas condicionais.

Dificuldades como estas apresentadas, encorajam o desenvolvimento de soluções que auxiliem no ensino e aprendizagem de programação de forma diferente ao atual modelo de ensino. Diversas abordagens de ensino são estudadas para facilitar o aprendizado dos alunos, algumas delas são: gamificação, programação imperativa, programação funcional e etc. Neste trabalho, é abordado o uso da gamificação na construção de uma ferramenta de apoio ao ensino e aprendizagem de programação desenvolvida com base nos requisitos identificados a partir da interação com ex alunos da disciplina de Algoritmos de Programação de Computadores, ofertada pela Universidade de Brasília, campus Gama.

Segundo BARATA et al. (2013), jogos bem projetados representam bons motivadores, uma vez que passa a sensação de satisfação e recompensa fazendo com que os jogadores persistam e fiquem engajados em realizar suas missões. Neste contexto, este poder motivacional dos jogos, passou a ser utilizado em outros contextos que não estão

relacionados diretamente aos jogos, uma prática conhecida atualmente como Gamificação do inglês *gamification* .

Para [Deterding et al. \(2011\)](#), o termo gamificação pode ser definido como a utilização de elementos e mecânica de jogos em contextos não relacionados a jogos. De acordo com [Baruque \(2015\)](#) a utilização destes elementos tornam tarefas reais em atividades mais atrativas e lúdicas e, conseqüentemente, aumentam a motivação e engajamento. Há uma grande variedade de ambientes que possuem elementos semelhantes a características de jogos, muitos deles contendo: sistema de pontuação, feedbacks constantes e etc ([BARATA et al., 2013](#)). São exemplos de ambientes com características semelhantes a de jogos: Uri, Datacamp, Edx entre outras.

A aprendizagem baseada na gamificação, preocupa-se em utilizar de mecanismos de jogos não para o entretenimento, mas para o ensino. Os interessados no campo da gamificação trabalham para identificar o cenário e as condições que possam apoiar a integração de jogos aos ambientes de aprendizado. Vários cientistas e estudiosos no campo da gamificação apontaram uma diversidade de elementos de jogos que permitem que eles sejam utilizados como ferramentas de apoio ao aprendizado. Por exemplo: os jogos são bastante envolventes ([DICKY, 2005](#)) e motivadores ([PRENSKY, 2003](#)). Além destas características, jogos são excelentes fontes para se adquirir experiência que são difíceis de serem fornecidas por meio de instruções tradicionais ([ARENA; SCHWARTZ, 2014](#)).

Os ambientes online gamificados de apoio ao ensino podem fornecer diversas ferramentas, entre elas: classificações, batalhas, fóruns de discussões e etc. De forma a incentivar os usuários a participarem das atividades propostas. Durante as competições e batalhas, os estudantes têm a possibilidade de aprender com outros jogadores e comparar suas habilidades, tornando o aprendizado mais prazeroso ([COMBÉFIS; BERESNEVIČIUS; DAGIENĖ, 2016](#)).

## 1.2 Justificativa

De acordo com [Souza \(2009\)](#), a abordagem de ensino tradicional de programação, que é aquela onde o professor apresenta uma série de conceitos aos alunos e os mesmos têm a tarefa de entender como se aplicam na resolução de problemas, para a maioria dos estudantes, se revelam muito abstratas.

Em seu trabalho de conclusão de curso pela Universidade de Brasília, campus Gama (FGA), [Calixto \(2016\)](#) apresenta uma pesquisa baseada nos dados referentes aos índices de aprovação, trancamento e reprovação dos alunos na disciplina de computação básica (atualmente a disciplina recebe o nome de Algoritmos e Programação de Computadores, ofertada no campus Gama).

Os dados utilizados por [Calixto \(2016\)](#), são referentes a um total de 60 turmas e 3286 alunos matriculados nas disciplinas de Computação Básica entre os anos de 2009 e 2013. Segundo o autor, dos 3286 alunos matriculados, 1659 alunos (50,48%) foram reprovados, 262 alunos (8%) trancaram a disciplina e apenas 1364 alunos (41,5%) obtiveram aprovação. O autor ainda explica que esses resultados, quando comparados à média nacional de aprovação de alunos em disciplinas de programação são bem preocupantes, a taxa apresentada como média nacional é de cerca de 67% de aprovação, ou seja, a taxa de aprovação na FGA, é mais de 30% menor em relação a média apresentada (67%).

Contando com a cooperação dos funcionários da secretaria da FGA, obteve-se dados atuais referentes às aprovações, reprovações e trancamentos dos alunos entre os períodos 2017/1 e 2019/1. Ao analisar estes dados, notou-se que do total de alunos matriculados (2225 alunos divididos em 37 turmas) no decorrer de 5 semestres, 890 alunos (40%) foram reprovados, 112 alunos realizaram o trancamento (5%) e apenas 1223 alunos foram aprovados (54,96%). Estes dados são preocupantes e demonstram a atual situação do aprendizado dos alunos na disciplina de Algoritmos e Programação de Computadores na FGA.

Um estudo realizado pela Universidade Federal da Paraíba (UFPB) que analisou por seis períodos acadêmicos os índices de reprovação na disciplina de Introdução à programação, apontou que em nenhum dos seis períodos analisados houvera um índice de aprovação superior a 34%. Além disso, os índices de reprovação e trancamento da disciplina giraram em torno dos 64% e 6% respectivamente ([DANTAS, 2016](#)).

Um outro estudo, realizado na Faculdade de Educação Tecnológica do Estado do Rio de Janeiro (FAETERJ-Paracambi) por [Vieira, Junior e Vieira \(2015\)](#), envolvendo 663 alunos da disciplina de Algoritmos 1 mostrou que, destes alunos, apenas 511 cursaram a disciplina até o final, ou seja, cerca de 152 alunos desistiram da disciplina. Do total restante (511), apenas 30% foram diretamente aprovados (cerca de 153 alunos), 40% foram reprovados diretamente (204 alunos) e 30% foram para o exame final (recuperação,



153). Dos 153 alunos que foram para a recuperação, apenas 55% foram aprovados (84 alunos), nota-se então que, dos 663 alunos que foram matriculados na disciplina, apenas 288 alunos foram aprovados ao seu término, cerca de 43%.

## 1.3 Problema

Nas seções anteriores, apresentou-se alguns estudos realizados a respeito da situação encontrada durante as análises iniciais do ensino e aprendizagem de programação em matérias introdutórias, a partir destes panoramas, o presente trabalho visa demonstrar os resultados e passos dados na construção de uma ferramenta gamificada para auxiliar o ensino e aprendizagem de programação.

## 1.4 Objetivos

Tendo sido apresentado o contexto e problema, o objetivo geral do presente trabalho é demonstrar o desenvolvimento e resultados obtidos durante a construção da mesma.

Para alcançar o objetivo geral, foram definidos os seguintes objetivos específicos:

- Desenvolvimento da ferramenta gamificada com base nos requisitos;
- Definição do método de coleta de dados a respeito da utilização da ferramenta pelos usuários;
- Análise dos dados a respeito da experiência de uso.

## 2 Referencial teórico

referencial

### 2.1 Aprendizagem de programação

De acordo com [Koliver, Dorneles e Casa \(2004\)](#), disciplinas introdutórias de programação são, em sua maioria, problemáticas e costumam apresentar altos índices de desistência e reprovações. Um dos motivos para tal ocorrência se dá pela falta de preparo que se espera que alunos ingressantes nestas disciplinas possuam.

Disciplina de algoritmos são normalmente oferecidas no início da grade curricular dos cursos. Isso faz com que a maioria dos alunos cursantes sejam calouros que ainda estão acostumados com a forma “mecanizada” de ensino que os habitua a somente aplicar fórmulas sem qualquer tipo de análise mais profunda dos problemas ([KOLIVER; DORNELES; CASA, 2004](#)).

De acordo com [Borges \(2019\)](#), o modo tradicional de ensino (figura 1) não é suficiente para motivar os alunos a se interessarem por disciplinas de programação. Não fica claro para a maioria dos alunos, principalmente para aqueles não possuem nenhum tipo de conhecimento em informática, a importância de certos conteúdos.

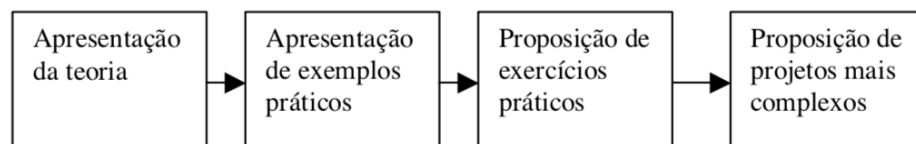


Figura 1 – Sequência de passos típicos na apresentação de uma disciplina

Fonte: ([BORGES, 2019](#))

Ao se apresentar uma nova linguagem de programação, é comum as aulas serem realizadas em laboratórios com recursos computacionais. Apesar dessas aulas apresentarem um formato diferenciado em relação ao modo tradicional, os professores não exploram a diversidade dos equipamentos disponíveis com práticas que apoiem o desenvolvimento de habilidades por parte dos estudantes ([BORGES, 2019](#)).

### 2.2 Gamificação

Jogos são uma construção humana que envolvem em seu contexto fatores sociais, culturais e econômicos ([TOLOMEI, 2017](#)). Como apresentado no livro “Gamificação na Educação” por [Silva et al. \(2014\)](#), a interação com *games*, apesar dos custos altos dos

consoles, foram ocupando cada vez mais o tempo das pessoas que notaram que os jogos poderiam ser boas fontes de prazer e entretenimento.

O notável crescimento do mercado dos *games* tem atraído diferentes olhares de estudiosos que se dedicam ao estudo de seu uso na educação, comunicação, marketing, psicologia, computação, entre outras áreas (SILVA et al., 2014).

O termo gamificação, segundo Silva et al. (2014), consiste na utilização de elementos de jogos em atividades que por natureza de sua criação não são jogos. Raposo e Dantas (2016) dizem que o objetivo da gamificação, consiste em resolver problemas práticos ou despertar interesse e engajar um público específico para a realização de uma determinada atividade.

Para Chou (2017), a gamificação é uma arte que é capaz de derivar elementos divertidos e envolventes encontrados em jogos e utiliza-los em outras atividades. Para o autor, o foco deve estar centrado no ser humano e na sua motivação.

### 2.2.1 Gamificação na Educação

Embora o termo gamificação tenha sido apresentado pela primeira vez em 2010, a idéia de se utilizar elementos de jogos em atividades que não são jogos, tem sido utilizado há muito tempo. Na educação de crianças, por exemplo, as mesmas podiam ter seus esforços e trabalhos reconhecidos por meio de estrelinhas ou outros tipos de recompensas dadas por seus educadores, como explica Silva et al. (2014).

De acordo com Silva et al. (2014), no Brasil existem diversas instituições públicas e privadas que apoiam o desenvolvimento e uso de ambientes gamificados. A exemplo disto, o Ministério da Educação visa fornecer suporte para o ambiente gamificado de apoio ao ensino *Geekie games* que possibilita aos estudantes se prepararem para o Exame Nacional de Ensino Médio (ENEM). Os resultados da utilização da ferramenta, segundo Silva et al. (2014), foram considerados positivos e o Ministério da Educação levanta a possibilidade de estender o uso da ferramenta gamificada para outros sistemas de avaliação.

Segundo Lee e Hammer (2011), somente a utilização de elementos de jogos no ensino não resolve a falta de empatia no processo de aprendizagem dos alunos uma vez que a utilização de mecanismos como por exemplo o sistema de pontuação já estão presentes no cotidiano escolar há anos. De acordo com as autoras, a utilização de elementos e característica de jogos devem provocar impactos tanto emocionais quanto sociais nos indivíduos para que eles tenham um aprendizado efetivo.

## 2.3 Framework Octalysis

No livro "*Actionable Gamification – Beyond Points, Badges, and Leaderboards*", Chou (2017) apresenta uma série de características que, segundo ele, atraem e motivam pessoas a tomarem decisões e a realizarem determinadas atividades. Chou (2017) ressalta ainda que essas características estão presentes na maioria dos jogos bem sucedidos.

Ao estudar uma variedade de técnicas de jogos que fazem com que os mesmos sejam tão atraentes, Chou (2017) notou que estas técnicas impulsionam os jogadores de maneiras diferentes. Algumas estimulam os jogadores a permanecerem no jogo por meio da inspiração e capacitação, outras estimulam por meio da manipulação e obsessão.

Com o aprofundamento de sua pesquisa e a partir da observação de como as técnicas de jogos influenciam nos jogadores, Chou (2017) propôs uma estrutura de design de gamificação que foi apelidada de *Framework Octalysis*.

O *Framework Octalysis* une elementos de *design* de jogos com a psicologia objetivando motivar os envolvidos no processo de gamificação a realizarem determinadas ações. O modelo de Chou (2017) apresenta oito eixos principais que são chamados de *core drives*, cada *core* apresenta características que por sua vez apresentam técnicas que influenciam diretamente os jogadores. Nas subseções a seguir, é apresentado cada um dos oito *core drives* definidos por Yu-Kai Chou.

### 2.3.1 Significado épico e chamado

Significado épico e chamado, segundo Chou (2017), é um *core* que provoca a sensação no jogador de que ele está envolvido em algo grande e que foi especialmente escolhido para executar uma determinada ação. Ao realizar uma contribuição para a plataforma Wikipedia, por exemplo, as pessoas acreditam que estão ajudando a proteger e disseminar algo que é maior que elas, o conhecimento humano (CHOU, 2017). O próprio autor relata em seu livro a experiência que teve com a criação de uma página na Wikipedia com informações de sua empresa, página criada por ele permaneceu no ar por pouco minutos. Vários contribuintes da Wikipedia relataram que a página não era suficiente para merecer estar na Wikipedia, vários membros concordaram com a insignificância da página e, após aproximadamente dez minutos a página foi retirada do ar. Para Chou (2017), isso ocorreu devido ao fato de as pessoas que constituíam a "comunidade Wikipedia", ao realizarem o trabalho voluntário de inspecionar os conteúdos da plataforma, sentirem que estão fazendo parte de algo que é maior que elas e que de certa forma estão protegendo o conhecimento humano.

Em seguida são apresentadas algumas técnicas de "Significado épico e chamado" segundo Chou (2017). Demais técnicas podem ser apreciadas em seu livro presentes no livro "*Actionable Gamification – Beyond Points, Badges, and Leaderboards*".

- Narrativa: A maioria dos jogos iniciam com uma história que mostra aos jogadores o contexto sobre o qual está inserido o jogo e o quão importante o jogador é para solucionar os desafios propostos. Uma das maneiras mais eficazes em utilizar este núcleo é por meio de uma narrativa envolvente.
- Herói da humanidade: Em seu livro, [Chou \(2017\)](#) apresenta exemplos que demonstram a natureza desta técnica. A empresa "*Tom's Shoes*", por exemplo, envia um par de sapatos para uma criança em país de terceiro mundo sempre que um de seus clientes realiza um novo pedido. O site "*Free Rice*" doa dez grãos de arroz para cada resposta correta dadas para as perguntas educacionais postadas no site. Mecanismos como este, passam a sensação de "heroísmo" aos envolvidos, fazendo com que os mesmos se sintam motivados em participar das atividades propostas;
- Sorte de iniciante;
- Criança predestinada;
- Almoço grátis;
- Elitismo;
- Significado superior.

### 2.3.2 Desenvolvimento e Conquista

Desenvolvimento e conquista é o segundo *core* do *framework* e está relacionado à motivação interna de cada pessoa em desenvolver suas habilidades e superar desafios. Neste núcleo, o desafio para se conquistar, por exemplo, um troféu ou distintivo é muito importante, uma vez que conquistar estes artefatos sem um desafio se torna insignificante. De acordo com [Chou \(2017\)](#), essa é a unidade mais fácil de ser projetada.

Alguns exemplos de técnicas utilizadas neste *core* são apresentadas em seguida.

- Pontos;
- Lista de desafios;
- Luta contra chefes;
- Tutorial passo a passo;
- Prêmios;
- Barra de progresso;
- Medalhas;

- Emblemas;
- Ranking.

### 2.3.3 Empoderamento e feedback

Empoderamento e feedback é o terceiro *core* e está ligado ao uso de táticas visando promover a realização pessoal do usuário de forma a aumentar seu potencial. Geralmente é expresso quando os usuários se envolvem em um processo criativo onde os mesmos descobrem coisas novas e, a partir disso, passam a tentar combinações novas, recebendo *feedbacks* de acordo com os resultados destas novas combinações.

Alguns exemplos de técnicas utilizadas neste *core* são apresentadas em seguida.

- Controle em tempo real;
- Desbloqueio de marcos;
- Combinações em cadeia;
- *Feedback* instantâneo;
- *Boosters*;
- Autonomia;
- Percepção de escolhas.

### 2.3.4 Propriedade e posse

Propriedade e posse é o quarto *core*. É onde os usuários são motivados a permanecerem nas atividades por sentirem que possuem controle sobre alguma coisa. De acordo com Chou (2017), quando uma pessoa sente que tem propriedade sobre alguma coisa, a atitude dela é de querer aumentar e melhorar o que possui. Se uma pessoa gasta muito tempo personalizando seu avatar, automaticamente sente mais propriedade e controle sobre ele ou quando ela é recompensada com uma moeda virtual ou bens virtuais, a postura dela é realizar mais atividades objetivando receber e acumular mais bens.

Alguns exemplos de técnicas utilizadas neste *core* são apresentadas em seguida.

- Bens virtuais;
- Construir a partir do zero;
- Coleções;

- Avatares;
- Curva de aprendizado;
- Proteção;
- Monitoramento.

### 2.3.5 Influência e relacionamento social

Influência e relacionamento social é o quinto *core* e envolve todos os elementos e características sociais que motivam as pessoas. São alguns destes elementos: aceitação social, competição, feedback social e até inveja. Em seu livro, [Chou \(2017\)](#) diz que, quando vemos um amigo que possui uma ótima habilidade em alguma coisa, logo nos sentimos motivados a alcançar o mesmo. Quando alguém nos fala que alcançou um determinado nível em um jogo, por exemplo, a nossa reação, na maioria das vezes, é de querer superar a pessoa e, portanto nos engajamos para realizar tal feito.

Alguns exemplos de técnicas utilizadas neste *core* são apresentadas em seguida.

- Amizades;
- Reconhecimento social;
- Atividades em grupo;
- Orientações.

### 2.3.6 Escassez e impaciência

A escassez e a impaciência é o sexto *core* e consiste na urgência de se obter algo só porque é raro, exclusivo ou inatingível. De acordo com [Chou \(2017\)](#), o facebook, por exemplo, na época em que foi lançado era exclusivo para estudantes de *Harvard*, depois foi aberto para outras Universidades de grande prestígio e finalmente, quando aberta para todos os públicos, universitários ou não, houve um grande aumento no número de pessoas que queriam participar da rede social pelo simples fato de que anteriormente não podiam.

Alguns exemplos de técnicas utilizadas neste *core* são apresentadas em seguida.

- Dinâmica de nomeação;
- Intervalos fixos;
- *Feedback paciente*;
- Fossos;

- Aceleradores;
- Contagem regressiva.

### 2.3.7 Imprevisibilidade e curiosidade

Imprevisibilidade e curiosidade é o sétimo *core* e exige o envolvimento constante, uma vez que não se tem nenhuma previsão do que acontecerá. Chou (2017) explica que, quando algo não se enquadra nos nossos ciclos regulares de reconhecimento de padrões, nosso cérebro entra em ação e passa a focar no inesperado, prendendo nossa atenção ao que pode vir a ocorrer. Yu-Kai Chou explica ainda que esse é o principal núcleo por trás dos vícios nos jogos de azar.

Alguns exemplos de técnicas utilizadas neste *core* são apresentadas em seguida.

- Escolha brilhante;
- Mini desafios;
- *Easter eggs*;
- Recompensas aleatórias;
- Travessuras;
- Maravilha óbvia;
- Recompensas repentinas.

### 2.3.8 Prevenção e perda

Prevenção e perda é o oitavo e último *core* do *framework* e está relacionado à motivação em se evitar que algo negativo aconteça. Exemplo: realizar alguma ação em um jogo para evitar que pontos sejam perdidos por falta de atividade.

Alguns exemplos de técnicas utilizadas neste *core* são apresentadas em seguida.

- Perda de progresso;
- Preguiça de status Quo;
- Carta escarlate;
- Sepulturas visuais.





os participantes. Os núcleos localizados na parte superior do octógono são considerados “motivadores positivos” enquanto os núcleos localizados na parte inferior do octógono são considerados "motivadores negativos". Os *cores* apresentados na parte superior são chamados de "*White hat*" e os inferiores são conhecidos como "*Black hat*". Na figura 3 é apresentado a divisão dos núcleos levando em consideração o tipo de motivador em que cada um é classificado.

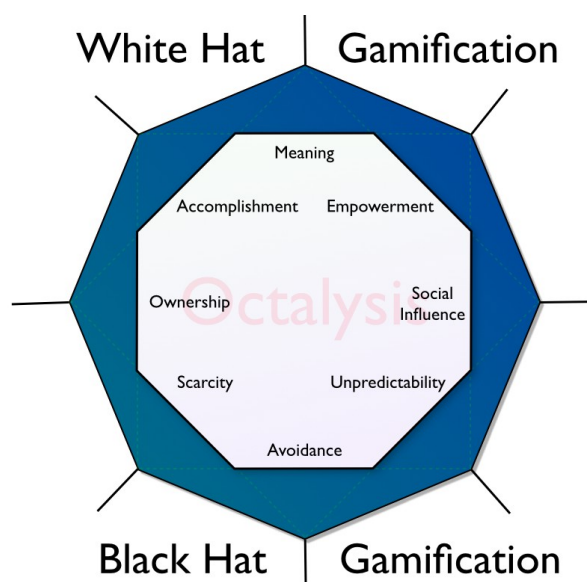


Figura 3 – Gamificação *White hat* e *Black hat*

Fonte: (CHOU, 2017)

Chou (2017) apresenta um exemplo do que vem a ser os núcleos "*White hat*" e "*Black hat*". Se algo é envolvente por permitir que se expresse a criatividade, faz com que os indivíduos envolvidos se sintam bem sucedidos e poderosos, com certeza há um *core* "*White hat*" sendo utilizado. Por outro lado, se os envolvidos realizam determinadas ações por não saber o que acontecerá logo em seguida ou está constantemente com medo que algo ruim possa acontecer, neste caso, há um *core* "*Black hat*" sendo utilizado.

Como apresentado por Chou (2017), não é porque alguma coisa é categorizada como "*black*" que ela seja ruim. Se utilizadas corretamente podem gerar resultados saudáveis e positivos. De acordo com Chou (2017), um profissional de gamificação deve sempre considerar todos os *core drive* objetivando obter resultados positivos e produtivos.

### 2.4.3 Níveis *Octalysis*

O *framework Octalysis* é dividido em vários níveis que, quanto mais alto forem, mais domínio por parte do “gamificador” é requerido (CHOU, 2017). Nas subseções que se seguem, são apresentadas as características relacionadas aos três primeiros níveis do *octalysis*.

### 2.4.3.1 Nível 1

Para ser aplicado, este nível deve levar em consideração pontos fortes e fracos de vários produtos, este primeiro método permite identificar que tipos de motivação são fracas para que se possa introduzir novas melhorias (CHOU, 2017).

O segundo método consiste na criação de uma nova experiência baseada nos *core drives* através de um processo sistemático.

Como citado em seções anteriores, neste nível sempre há a necessidade de se utilizar ao menos um *core drive*, caso não seja utilizado nenhum dos oito núcleos, a motivação é zero e nenhum progresso relativo à gamificação é alcançado (CHOU, 2017).

### 2.4.3.2 Nível 2

Este nível envolve todas as quatro fases da jornada de um jogador. Em seguida são apresentadas cada uma das quatro fases, bem como suas características.

- Descoberta: Experimentações e ganhos de experiências por parte dos jogadores;
- Entrada: Aprendizado das regras e ferramentas necessárias para jogar o *game*.
- Dia a dia: Jornada regular de ações e comportamentos que os jogadores devem realizar para completar os objetivos;
- Fim de jogo: Manter os jogadores motivados a continuarem jogando.

Chou (2017) argumenta que o motivo pelo qual as pessoas utilizam um produto no dia 1 muitas das vezes é bastante diferente do motivo de uso no dia 100. Por este motivo, é importante se planejar cada uma das quatro fases envolvidas na trajetória dos jogadores. Caso não haja nenhum *core drive* envolvido em alguma das fases, o jogador não possui uma razão para passar para a próxima fase e ele simplesmente desiste.

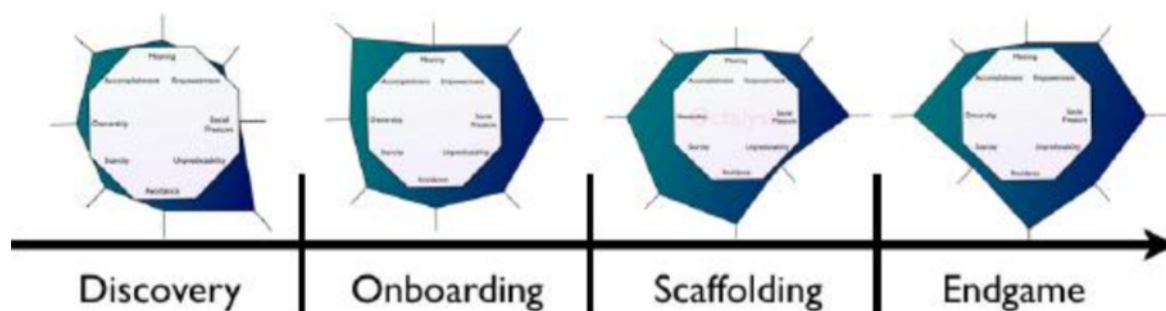


Figura 4 – As fases da jornada de um jogador.

Fonte: (CHOU, 2017)

Observando a figura 4, é possível notar os *core drives* mais proeminentes durante cada fase de experiência da jornada dos jogadores.

#### 2.4.3.3 Nível 3

Este nível está relacionado com a identificação dos perfis de jogadores e como eles se comportam quando estão envolvidos em jogos. Mesmo que seja praticamente impossível, este nível permite aos projetistas de gamificação buscar atender as necessidades e motivações dos mais diferentes perfis.

##### 2.4.3.3.1 *Design* de gamificação: Octalysis e Bartle

Embora a classificação dos perfis dos jogadores seja uma proposta criada por Bartle (1996), Chou (2017) apresenta um modelo que une os perfis apresentados por Bartle com os *core drives* apresentados no *framework Octalysis* de forma a criar um *design* de gamificação mais apropriado para cada tipo de perfil.

Como apresentado por Vianna et al. (2013) no livro “Gamification: reinventando empresas a partir de jogos” e com base nos estudos de Bartle (1996), existem quatro perfis de jogadores que possuem características e comportamentos específicos quando envolvidos em jogos. Cada um destes quatro perfis são apresentados na subseção a seguir.

#### 2.4.3.4 Perfis de Jogadores

##### 2.4.3.4.1 Predadores (*Killers*)

São aqueles jogadores que participam de uma competição pelo prazer de derrotar os adversários. O objetivo desse tipo de jogador é ser o melhor, não se importam com o que está em disputa. Possuem comportamentos agressivos e focados em conquistar a posição de liderança. Seus desejos de liderança se sobrepõem à cooperação.

##### 2.4.3.4.2 Realizadores (*Achievers*)

A motivação destes jogadores consiste na realização de todas as atividades que o jogo apresenta. Esses jogadores apreciam a sensação constante de vitória mesmo que os objetivos a serem alcançados não sejam tão relevantes ou significativos. Atuam de maneira cordialmente competitiva, mesmo que não estejam liderando algum placar. São caracterizados por se destacarem dos demais jogadores de maneira leal, isto é, por meio de suas próprias conquistas.

#### 2.4.3.4.3 Exploradores (*Explorers*)

Este é um grupo que se caracteriza pela busca e interesse em desvendar todas as possibilidades dos jogos. São indivíduos curiosos que se dedicam em estudar e desenvolver habilidade que lhes permitam solucionar os mais diversos desafios. Para este tipo de perfil, a trajetória é mais importante que a conquista.

#### 2.4.3.4.4 Socializadores (*Socializers*)

São, no geral, os indivíduos que enxergam o contato com jogos como um meio de interação social. Mais significativo que atingir os objetivos propostos ou realizar determinadas tarefas é a oportunidade de reforçar e estabelecer vínculos sociais. Os socializadores, em sua maioria, preferem jogos cooperativos que necessitam de trabalho conjunto. Estes representam 80% do total de jogadores.

#### 2.4.3.5 Relação entre *Core drives* e Perfis de jogadores

Como apresentado por [Chou \(2017\)](#) em seu livro "*Actionable Gamification – Beyond Points, Badges, and Leaderboards*", para cada perfil de jogador, existem um conjunto de *core drives* que os motivam a engajarem no desenvolvimento e realização de determinadas ações dentro de um jogo. Na tabela a seguir, são apresentados os *core drives* predominantes em cada perfil que, se utilizados de maneira correta, podem aumentar o interesse e engajamento dos perfis identificados na seção anterior.

Perfil	CD 1 - Significado épico e chamado	CD 2 - Desenvolvimento e Conquista	CD 3 - Empoderamento e feedback	CD 4 - Propriedade e posse	CD 5 - Influência e relacionamento social	CD 6 - Escassez e impaciência	CD 7 - Imprevisibilidade e curiosidade	CD 8 - Prevenção e perda
Predadores ( <i>Killers</i> )		x			x			
Realizadores ( <i>Achievers</i> )		x				x		
Exploradores ( <i>Explorers</i> )							x	
Socializadores ( <i>Socializers</i> )					x			

Tabela 1 – Principais *core drives* envolvidos em cada perfil de jogador

Fonte: ([CHOU, 2017](#))

Chou (2017) apresenta um gráfico relacionando os quatro perfis dos jogadores com os *core drives* característicos de cada um deles. O gráfico é apresentado na figura 5.

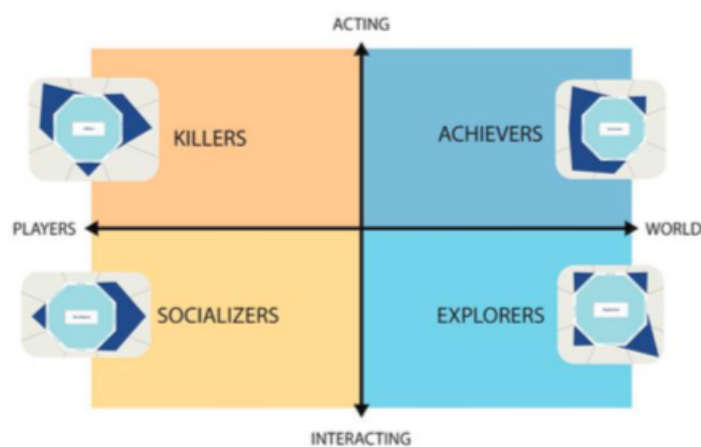


Figura 5 – Perfil de jogadores

Fonte: (CHOU, 2017)

Na figura 6, são apresentados os *core drives* mais utilizados em cada uma das quatro fases da jornada dos jogadores levando em consideração os quatro tipos de perfis identificados por Bartle (1996).

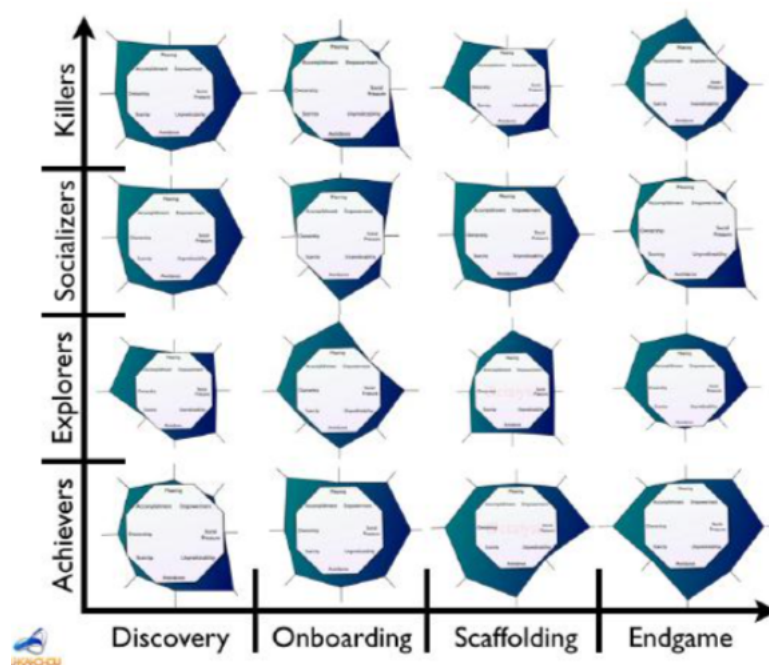


Figura 6 – *Core drives* mais utilizados de acordo com as fases e perfis de jogadores.

Fonte: (CHOU, 2017)

## 2.5 Modelagem de banco de dados

Existem diversas formas de se representar graficamente a forma como uma base de dados deverá estar consolidada em um banco de dados após sua implementação. Nesta seção, apresenta-se uma forma específica de modelagem de base de dados, bem como suas características.

### 2.5.1 Diagrama entidade-relacionamento

Segundo [Heuser \(1998\)](#), a abordagem mais utilizada e conhecida é a entidade-relacionamento (ER) onde o modelo de dados é geralmente representado graficamente através de um diagrama entidade-relacionamento (DER). Esta abordagem foi criada em 1976 por Peter Chen e é considerada como um padrão para a modelagem conceitual.

A abordagem entidade-relacionamento é baseada em dois principais pilares que são apresentados em seguida.

#### 2.5.1.1 Entidade

De acordo com [Heuser \(1998\)](#), uma entidade, no modelo conceitual, representa um conjunto de objetos da realidade modelada. Seu principal objetivo é modelar de forma abstrata um banco de dados, onde se tem interesse somente nos objetos sobre os quais deseja-se manter informações. No DER, uma entidade é representada por meio de um retângulo contendo o nome da entidade.

#### 2.5.1.2 Relacionamento

Como apresentado por [Heuser \(1998\)](#), o DER permite especificar as propriedades dos objetos que serão armazenados no banco de dados, como por exemplo o relacionamento/associação entre os objetos. No DER, um relacionamento é representado por meio de um losango que são ligados por linhas aos retângulos que representam as entidades que participam de um determinado relacionamento.

Na figura 7 é apresentado um exemplo simples de um diagrama entidade-relacionamento (DER). É possível notar no diagrama a existência de atributos e cardinalidade, estes dois são apresentados logo em seguida.

Os atributos correspondem às características/qualidades que descrevem uma entidade, são representados por elipses ou círculos acompanhados por seus respectivos nomes ([CARDOSO; MARA, 2013](#)).

As cardinalidades representam a restrição do número de objetos que podem participar do relacionamento. Na notação de Peter Chen, as cardinalidades são apresentadas próximas às ligações de relacionamento e são compostas da quantidade mínima e máxima

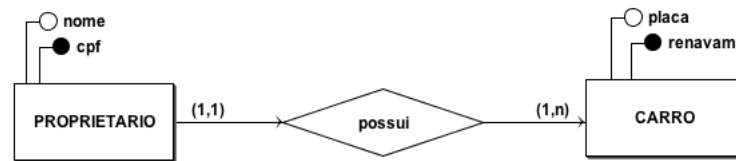


Figura 7 – Exemplo de DER

Fonte: Autor

de objetos que podem participar do relacionamento. Tais características podem ser vistas na figura 7 (CARDOSO; MARA, 2013).

### 2.5.2 Diagrama Lógico

O diagrama lógico leva em consideração as limitações e implementa recursos como adequação de padrão e nomenclatura, define as chaves primárias e estrangeiras, normalização, integridade referencial, entre outras. Para o modelo lógico deve ser criado levando em conta os exemplos de modelagem de dados criados no modelo conceitual (CARDOSO; MARA, 2013).

## 2.6 Core drives

core drives



## 3 Metodologia

Neste capítulo, são apresentados com detalhes o passo a passo realizado para a construção de uma ferramenta web de apoio ao ensino e aprendizagem de programação gamificada.

### 3.1 Requisitos

Tendo como base a pesquisa realizada na primeira parte deste trabalho, nesta seção, são apresentados os requisitos implementados na construção da ferramenta.

Como apresentado no referencial teórico deste trabalho, o *framework octalysis* é dividido em vários níveis. Neste trabalho, a solução tecnológica foi construída sobre o nível 1.

Na tabela a seguir, é apresentado uma comparação entre os requisitos levantados em relação aos implementados.

Identificador	Requisito	Implementado?
RF 01	CRUD de jogadores	Sim
RF 02	Ativar/desativar perfil de jogadores	Sim
RF 03	Rankeamento de jogadores	Sim
RF 04	CRUD de disciplinas	Não
RF 05	CRUD de turmas	Não
RF 06	CRUD de conteúdos para estudo	Sim
RF 07	CRUD de desafios	Sim
RF 08	Sistema de recompensas (pontuação)	Sim
RF 09	Dashboard pessoal de desempenho (jogadores)	Sim
RF 10	CRUD de professores de disciplinas	Não
RF 11	Painel de acompanhamento de desempenho dos alunos (professores)	Não
RF 12	Narrativa temática	Sim
RF 13	Personagens e guerreiros da narrativa	Sim
RF 14	Lógica de progressão de jogadores ( <i>levels</i> )	Sim
RF 15	CRUD de linguagens de programação	Sim
RF 16	Desafios entre jogadores (individual)	Sim
RF 17	CRUD de questões	Sim
RF 18	Desafios entre times/grupos	Não
RF 19	Torneios/campeonatos entre turmas	Não

Tabela 2 – Requisitos implementados

Fonte: Autor

## 4 Características da ferramenta

Neste capítulo são apresentados os elementos característicos da ferramenta como: personagens, narrativa, temática entre outros elementos que foram pensados como forma de gamificar as atividades de aprendizagem de programação.

### 4.1 Narrativa

Com o objetivo de aumentar o engajamento dos estudantes/jogadores, fora desenvolvida uma história que se passa em um mundo onde criaturas (Orcs) invadem o vilarejo do jogador que, motivado pelo desejo de vingança e tendo sido escolhido entre uma legião de outros guerreiros, dá início à jornada onde o mesmo deve cumprir com desafios como: quiz, desafiar outros jogadores entre outras atividades que dão ao jogador pontos que podem ser trocados por itens ou serem acumulados a fim de obter boas posições nos *rankings*.

### 4.2 Personagens

Ao jogador, após a realização do primeiro *login* na ferramenta, são apresentados seis personagens com diferentes características, onde o jogador deve selecionar um único personagem para iniciar a jornada de aprendizado. Ao selecionar um determinado personagem, não é permitido ao jogador/estudante modificá-lo no decorrer do uso da ferramenta.

Os personagens são apresentados a seguir.

#### 4.2.1 Aisha



Figura 8 – Personagem Aisha

Fonte: Autor

### 4.2.2 Voxter

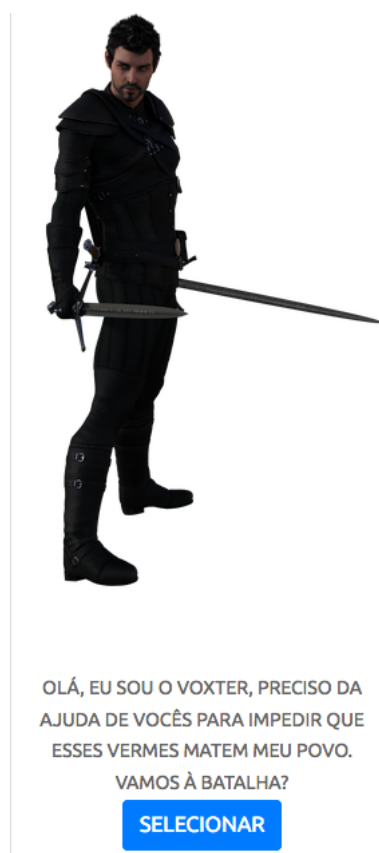


Figura 9 – Personagem Voxter

Fonte: Autor

### 4.2.3 Lince

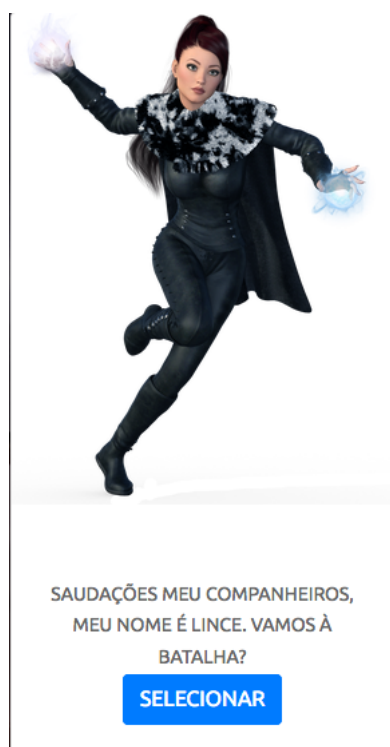


Figura 10 – Personagem Lince

Fonte: Autor

#### 4.2.4 Amazona



Figura 11 – Personagem Amazona

Fonte: Autor

### 4.2.5 Tymer



MEU NOME É TYMER, VAMOS  
EXPULSAR TODOS ESSE MALDITOS?.

SELECIONAR

Figura 12 – Personagem Tymer

Fonte: Autor

### 4.2.6 Scar



Figura 13 – Personagem Scar

Fonte: Autor

## 4.3 Questões

Existem diferentes tipos de questões que podem ser solucionados pelo jogador. A correção destas, são feitas automaticamente pelo sistema de acordo com respostas pré cadastradas na base de dados.

Os tipos de questões são:

- Verdadeiro ou falso;
- Lacunas;
- Múltipla escolha

Cada resposta contabilizada como correta, são adicionados 10 pontos à pontuação total do jogador. Não há limites de tentativas de solucionar as questões.

Para cada tentativa de resposta julgada como incorreta, é apresentado a alternativa correta para aquela questão.



## 4.4 Levels

A medida que o jogador conquista maiores pontuações, é desbloqueado o novo level que consiste na liberação de novos conteúdos mais avançados em relação aos anteriores. No total, existem 12 levels, os conteúdos abordados em cada um deles são apresentado logo a seguir.

<b>Level</b>	<b>Conteúdo</b>
01	Algoritmos
02	Linguagem de Programação
03	Tipos de dados
04	Entrada e saída de dados
05	Operações primitivas
06	Variáveis e expressões
07	Estruturas de decisão
08	Laços de repetição
09	Vetores
10	Matrizes
11	Funções e Procedimentos
12	Arquivos

Tabela 3 – Conteúdos por *level*

Fonte: Autor

Os levels são compostos por questões, como as apresentadas na seção anterior.

## 4.5 Desafios

### 4.5.1 Jogador contra Jogador

### 4.5.2 Desafio individual

Os desafios individuais consistem em resolver diferentes tipos de desafios pré cadastrados no sistema que, uma vez solucionados e avaliados pelo administrador do sistema e, tendo sido julgado por este como correto, é adicionado à pontuação total do jogador mais 100 pontos. Os desafios variam de simples respostas até implementação de códigos.

Fonte: Autor

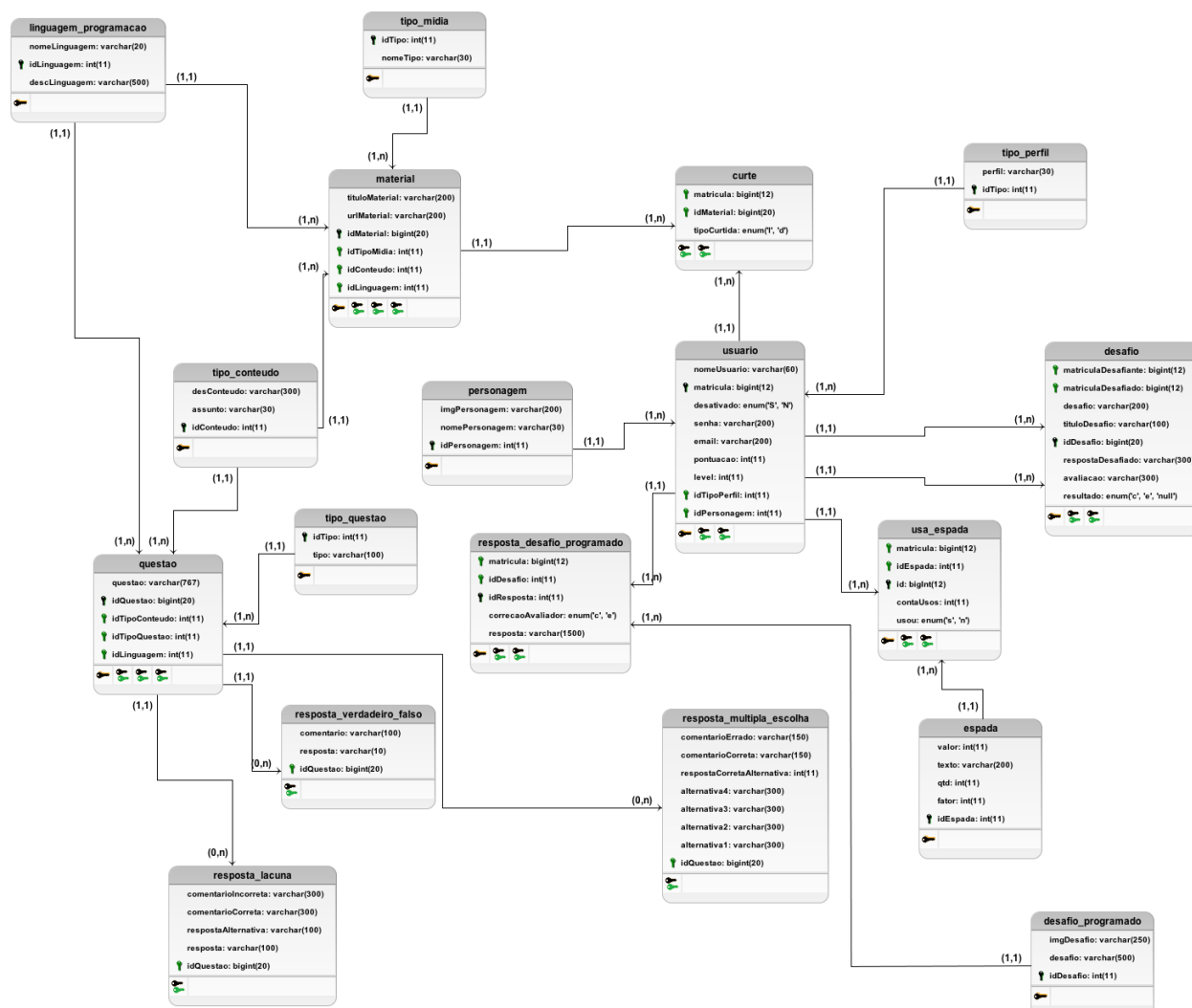


Figura 15 – DL da ferramenta

Fonte: Autor

## Referências

- ALMEIDA, E. et al. Ambap: Um ambiente de apoio ao aprendizado de programação. 09 2019. Citado na página 13.
- ARENA, D. A.; SCHWARTZ, D. L. Experience and explanation: Using videogames to prepare students for formal instruction in statistics. *Journal of Science Education and Technology*, v. 23, n. 4, p. 538–548, Aug 2014. ISSN 1573-1839. Disponível em: <<https://doi.org/10.1007/s10956-013-9483-3>>. Citado na página 14.
- BARATA, G. et al. Engaging engineering students with gamification. In: *2013 5th International Conference on Games and Virtual Worlds for Serious Applications (VS-GAMES)*. [S.l.: s.n.], 2013. p. 1–8. Citado 2 vezes nas páginas 13 e 14.
- BARTLE, R. Hearts, clubs, diamonds, spades: Players who suit muds. *Journal of MUD research*, v. 1, n. 1, p. 19, 1996. Citado 2 vezes nas páginas 27 e 29.
- BARUQUE, A. B. e L. Gamificação aplicada na graduação em jogos digitais. *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação - SBIE)*, v. 26, n. 1, p. 677, 2015. ISSN 2316-6533. Disponível em: <<https://www.br-ie.org/pub/index.php/sbie/article/view/5338>>. Citado na página 14.
- BORGES, M. Avaliação de uma metodologia alternativa para a aprendizagem de programação. 09 2019. Citado na página 17.
- CALIXTO, G. de L. Fatores relacionados ao ensino de programação: Uma análise das disciplinas introdutórias em cursos de engenharia de software. p. 1–82, 2016. Citado na página 15.
- CARDOSO, G. C.; MARA, V. C. *Sistema de banco de dados: Uma abordagem introdutória e aplicada*. 1. ed. [S.l.: s.n.], 2013. Citado 2 vezes nas páginas 30 e 31.
- CHOU, Y.-k. *Actionable gamification: Beyond points, badges, and leaderboards*. [S.l.]: Octalysis Media, 2017. Citado 12 vezes nas páginas 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28 e 29.
- COMBÉFIS, S.; BERESNEVIČIUS, G.; DAGIENĖ, V. Learning programming through games and contests: Overview, characterisation and discussion. *Olympiads in Informatics*, v. 10, 2016. Disponível em: <[https://ioinformatics.org/journal/v10\\_2016\\_39\\_60.pdf](https://ioinformatics.org/journal/v10_2016_39_60.pdf)>. Citado na página 14.
- DAI, K.; ZHAO, Y.; CHEN, R. Research and practice on constructing the course of programming language. In: . [S.l.: s.n.], 2010. p. 2033 – 2038. Citado na página 13.
- DANTAS, E. R. e V. O desafio da serpente - usando gamification para motivar alunos em uma disciplina introdutória de programação. *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação - SBIE)*, v. 27, n. 1, p. 577, 2016. ISSN 2316-6533. Disponível em: <<https://www.br-ie.org/pub/index.php/sbie/article/view/6739>>. Citado na página 15.

- DETERDING, S. et al. From game design elements to gamefulness: Defining "gamification". In: *Proceedings of the 15th International Academic MindTrek Conference: Envisioning Future Media Environments*. New York, NY, USA: ACM, 2011. (MindTrek '11), p. 9–15. ISBN 978-1-4503-0816-8. Disponível em: <http://doi-acm-org.ez54.periodicos.capes.gov.br/10.1145/2181037.2181040>. Citado na página 14.
- DICKEY, M. D. Engaging by design: How engagement strategies in popular computer and video games can inform instructional design. *Educational Technology Research and Development*, v. 53, n. 2, p. 67–83, Jun 2005. ISSN 1556-6501. Disponível em: <https://doi.org/10.1007/BF02504866>. Citado na página 14.
- HEUSER, C. A. *Projeto de banco de dados*. 4. ed. [S.l.: s.n.], 1998. Citado na página 30.
- KOLIVER; DORNELES; CASA. p. 949–960, 2004. Citado na página 17.
- LEE, J.; HAMMER, J. Gamification in education: What, how, why bother? *Academic Exchange Quarterly*, v. 15, p. 1–5, 01 2011. Citado na página 18.
- PEREIRA, R.; COSTA, C. J.; APARICIO, J. T. *Gamification to support programming learning*. [S.l.], 2017. 1-6 p. Citado na página 13.
- PRENSKY, M. Digital game-based learning. *Comput. Entertain.*, ACM, New York, NY, USA, v. 1, n. 1, p. 21–21, out. 2003. ISSN 1544-3574. Disponível em: <http://doi-acm-org.ez54.periodicos.capes.gov.br/10.1145/950566.950596>. Citado na página 14.
- RAPOSO, E. H. S.; DANTAS, V. O desafio da serpente-usando gamification para motivar alunos em uma disciplina introdutória de programação. In: *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)*. [S.l.: s.n.], 2016. v. 27, n. 1, p. 577. Citado na página 18.
- SILVA, A. R. L. da et al. *Gamificação na educação*. [S.l.]: Pimenta Cultural, 2014. Citado 2 vezes nas páginas 17 e 18.
- SOUZA, C. M. de. Visualg-ferramenta de apoio ao ensino de programação. *Revista Eletrônica TECCEN*, v. 2, n. 2, p. 01–09, 2009. Citado na página 15.
- THAIS, H. et al. Utilizando programação funcional em disciplinas introdutórias de computação. p. 3000–69077, 07 2002. Citado na página 13.
- TOLOMEI, B. A gamificação como estratégia de engajamento e motivação na educação. *EAD EM FOCO*, v. 7, n. 2, 2017. ISSN 2177-8310. Disponível em: <http://eademfoco.cecierj.edu.br/index.php/Revista/article/view/440>. Citado na página 17.
- VIANNA, Y. et al. *Como reinventar empresas a partir de jogos*. [S.l.]: MJV Press, 2013. Citado na página 27.
- VIEIRA, C. E. C.; JUNIOR, J. A. T. de L.; VIEIRA, P. de P. Dificuldades no processo de aprendizagem de algoritmos: uma análise dos resultados na disciplina de al1 do curso de sistemas de informação da faeterj–campus paracambi. *Cadernos UniFOA*, v. 10, n. 27, p. 5–15, 2015. Citado na página 15.

## Apêndices

# APÊNDICE A – Primeiro Apêndice

Texto do primeiro apêndice.

## APÊNDICE B – Segundo Apêndice

Texto do segundo apêndice.



# Anexos

## ANEXO A – Primeiro Anexo

Texto do primeiro anexo.

## ANEXO B – Segundo Anexo

Texto do segundo anexo.