



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, 2020-39 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, antecidas 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 – Exemplo de DER	20
Figura 3 – Personagem Aisha	23
Figura 4 – Personagem Voxter	24
Figura 5 – Personagem Lince	25
Figura 6 – Personagem Amazona	26
Figura 7 – Personagem Tymer	27
Figura 8 – Personagem Scar	28
Figura 9 – DER da ferramenta	30
Figura 10 – DL da ferramenta	31

Lista de tabelas

Tabela 1 – Requisitos implementados	21
---	----

Sumário

1	INTRODUÇÃO	13
1.1	Contexto	13
1.2	Justificativa	15
1.3	Problema	16
1.4	Objetivos	16
2	REFERENCIAL TEÓRICO	17
2.1	Aprendizagem de programação	17
2.2	Gamificação	17
2.2.1	Gamificação na Educação	18
2.3	Modelagem de banco de dados	19
2.3.1	Diagrama entidade-relacionamento	19
2.3.1.1	Entidade	19
2.3.1.2	Relacionamento	19
2.3.2	Diagrama Lógico	20
2.4	Core drives	20
3	METODOLOGIA	21
3.1	Requisitos	21
4	CARACTERÍSTICAS DA FERRAMENTA	22
4.1	Narrativa	22
4.2	Personagens	22
4.2.1	Aisha	22
4.2.2	Voxter	24
4.2.3	Lince	25
4.2.4	Amazona	26
4.2.5	Tymer	27
4.2.6	Scar	28
4.3	Questões	28
4.4	Níveis	29
4.5	Desafios	29
4.5.1	Jogador contra Jogador	29
4.5.2	Desafio individual	29
5	SOLUÇÃO TÉCNICA	30

5.1	Modelagem de dados	30
	REFERÊNCIAS	32
	APÊNDICES	34
	APÊNDICE A – PRIMEIRO APÊNDICE	35
	APÊNDICE B – SEGUNDO APÊNDICE	36
	ANEXOS	37
	ANEXO A – PRIMEIRO ANEXO	38
	ANEXO B – SEGUNDO ANEXO	39

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.

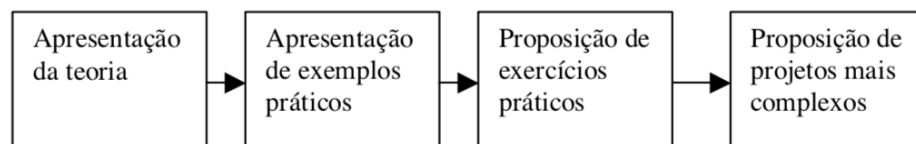


Figura 1 – Sequência de passos típicos na apresentação de uma disciplina

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 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.

2.3.1 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.

2.3.1.1 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.

2.3.1.2 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 2 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

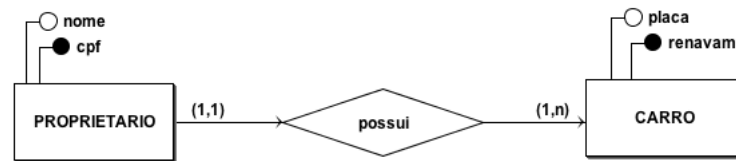


Figura 2 – Exemplo de DER

Fonte: Autor

de objetos que podem participar do relacionamento. Tais características podem ser vistas na figura 2 (CARDOSO; MARA, 2013).

2.3.2 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).

2.4 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 1 – 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, foi 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 3 – Personagem Aisha

Fonte: Autor

4.2.2 Voxter

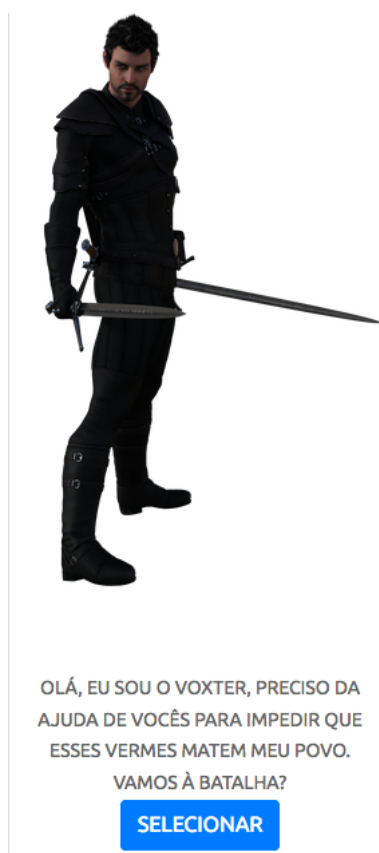


Figura 4 – Personagem Voxter

Fonte: Autor

4.2.3 Lince

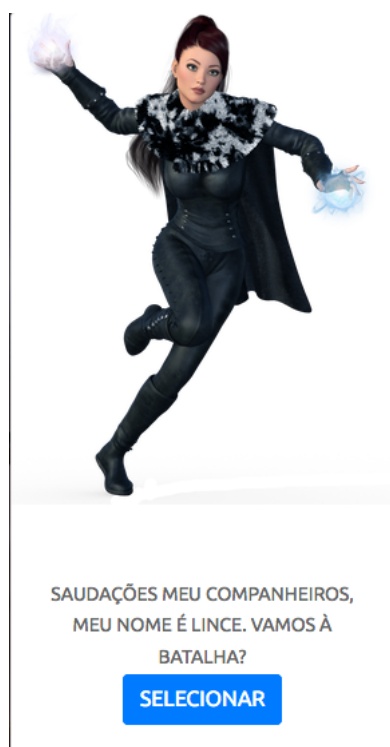


Figura 5 – Personagem Lince

Fonte: Autor

4.2.4 Amazona



Figura 6 – Personagem Amazona

Fonte: Autor

4.2.5 Tymer



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

SELECIONAR

Figura 7 – Personagem Tymer

Fonte: Autor

4.2.6 Scar



Figura 8 – 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 Níveis

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.

5 Solução técnica

5.1 Modelagem de dados

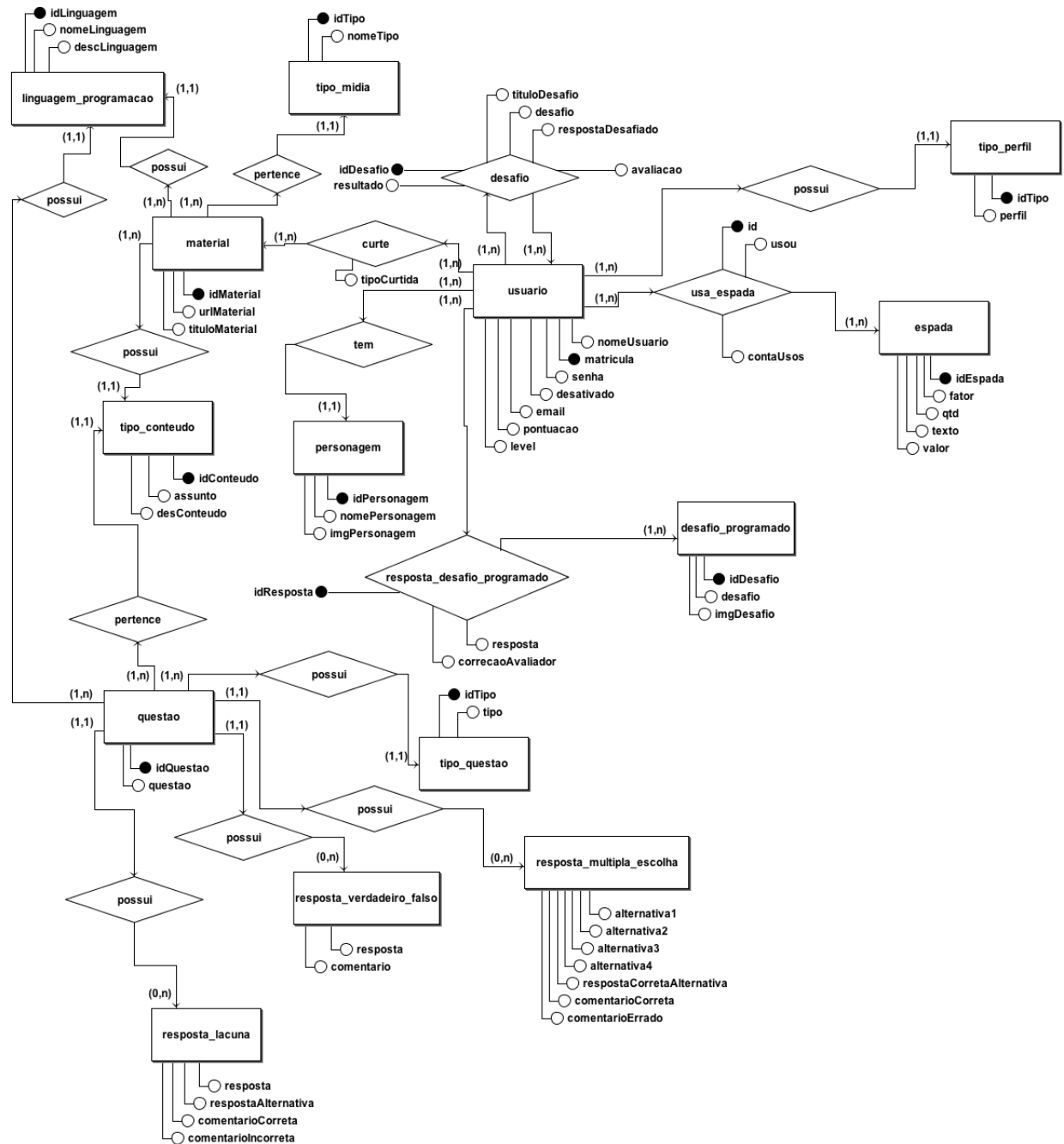


Figura 9 – DER da ferramenta

Fonte: Autor

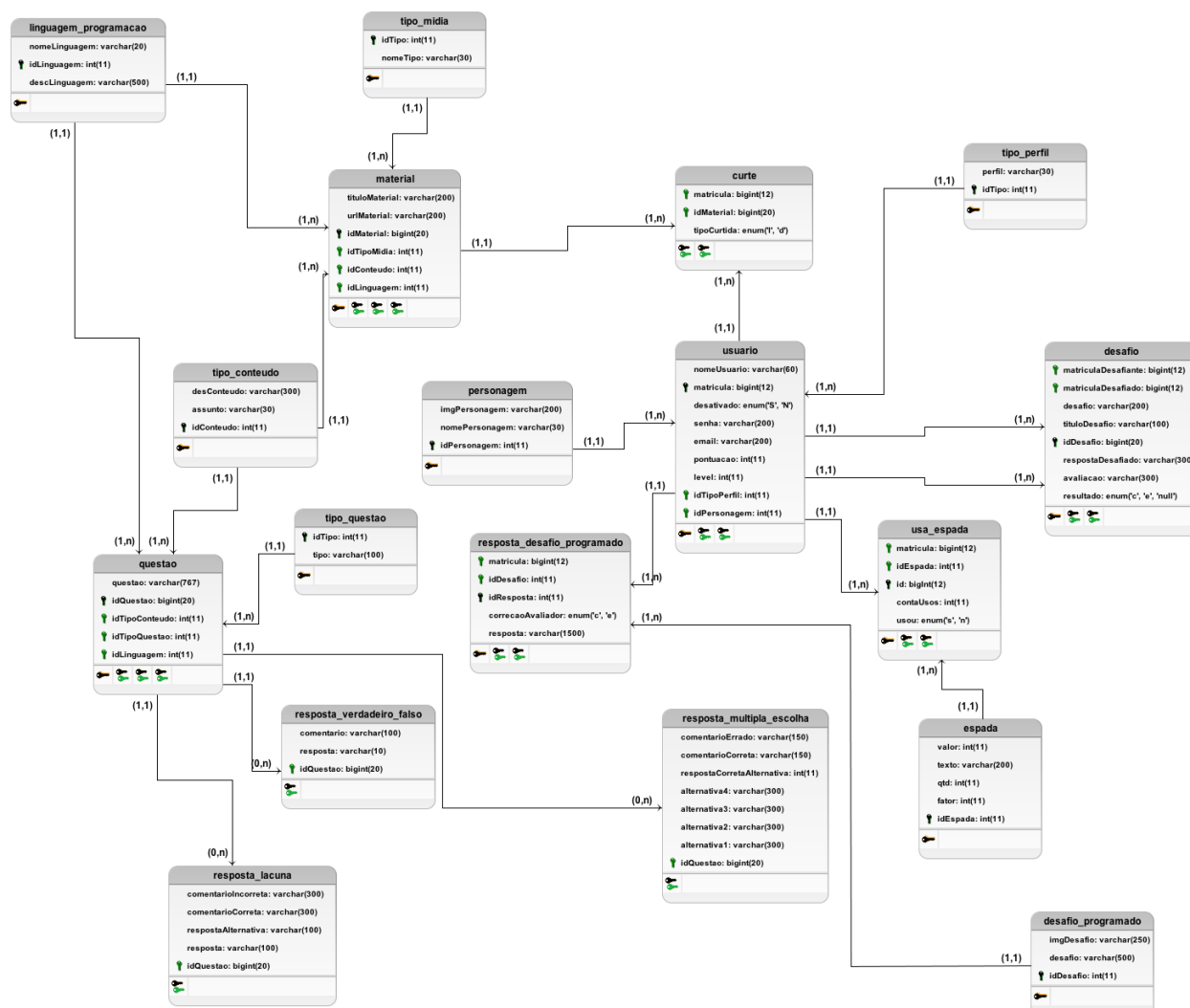


Figura 10 – DL da ferramenta

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.
- 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 19 e 20.
- CHOU, Y.-k. *Actionable gamification: Beyond points, badges, and leaderboards*. [S.l.]: Octalysis Media, 2017. Citado na página 18.
- 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>. 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 19.
- 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.
- 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.