



UNIVERSIDADE ESTADUAL DO NORTE DO PARANÁ
CAMPUS LUIZ MENEGHEL

GUILHERME BUENO PERESINI

**ANÁLISE COMPARATIVA DE FERRAMENTAS
GAMIFICADAS PARA O AUXÍLIO DE
APRENDIZAGEM DA DISCIPLINA DE
PROGRAMAÇÃO**

Bandeirantes
2018

GUILHERME BUENO PERESINI

**ANÁLISE COMPARATIVA DE FERRAMENTAS
GAMIFICADAS PARA O AUXÍLIO DE
APRENDIZAGEM DA DISCIPLINA DE
PROGRAMAÇÃO**

Trabalho de Conclusão de Curso apresentado à
Universidade Estadual do Norte do Paraná
Campus Luiz Meneghel com requisito para
obtenção do grau de bacharel em Sistemas de
Informação.

Orientador: Prof^o. Me. Fabio Moreno

Bandeirantes

2018

GUILHERME BUENO PERESINI

**ANÁLISE COMPARATIVA DE FERRAMENTAS
GAMIFICADAS PARA O AUXÍLIO DE
APRENDIZAGEM DA DISCIPLINA DE
PROGRAMAÇÃO**

Trabalho de Conclusão de Curso apresentado à
Universidade Estadual do Norte do Paraná
Campus Luiz Meneghel com requisito para
obtenção do grau de bacharel em Sistemas de
Informação.

Orientador: Prof^o. Me. Fabio Moreno

COMISSÃO EXAMINADORA

Prof^o. Me. Fabio Moreno
UENP – *Campus* Luiz Meneghel

Prof. Me. Fabio de Sordi Junior
UENP – *Campus* Luiz Meneghel

Prof. Dr. Mauricio Massaru Arimoto
UENP – *Campus* Luiz Meneghel

Bandeirantes, 18 de Julho de 2018

Dedico este trabalho à minha família, e a minha namorada Jaqueline, sem os quais eu não teria tido forças para realizá-lo.

AGRADECIMENTOS

Agradeço a Deus primeiramente, por estar sempre ao meu lado iluminando minhas decisões e me dando força nas horas mais difíceis na minha jornada.

Ao Prof. Me. Fabio Carlos Moreno, pela orientação e ajuda no desenvolvimento do trabalho.

Aos professores Dr. Mauricio Massaru Arimoto e Me. Fábio di Sordi Junior, pelas sugestões que foram de grande importância para o aprimoramento do trabalho.

Principalmente quero agradecer quem sempre esteve comigo meu pai, minha mãe, meu irmão, Ovidio, Irani, Gustavo, e não podendo esquecer a força e companhia que ganhei da minha namorada Jaqueline por não terem poupado esforços para incentivar a minha formação acadêmica.

“Tudo posso naquele que me fortalece”

RESUMO

O crescimento e o grande uso da tecnologia nos dias atuais faz com que cursos e faculdades voltadas à área tecnológica tenham uma grande procura, pois o mercado de trabalho tem uma grande demanda nesta área. Um grande problema que ocorre nas faculdades de tecnologia é o grande índice de evasão de alunos logo no início do curso. Muitos acadêmicos que ingressam à cursos da área de informática, encontram dificuldade para a assimilação do conteúdo ministrado na disciplina de programação, sendo que em muitas vezes, não conseguem assimilar esta disciplina, então optando pela desistência do curso. E para tentar encontrar alternativas para este fato, este trabalho apresenta um estudo comparativo entre algumas ferramentas que propõem um ambiente de programação gamificado, com o intuito de melhorar a aprendizagem da disciplina de programação de uma forma mais dinâmica e competitiva, bem como auxiliar os discentes a buscar conhecimentos externos aos aplicados na sala de aula. As ferramentas utilizadas foram o Robocode, CodeCombat, CodeCademy e Codingame, as quais possuem o mesmo objetivo que é a programação por meio de desafios. Através de um método de comparação, foi possível analisar as ferramentas e seus elementos de gamificação. Com os estudos e comparações realizados no trabalho, foi possível determinar qual ferramenta possui mais itens gamificados em relação as demais, e também qual poderá ser mais adequada para aprendizagem de lógica de programação.

Palavras-chave: programação, gamificação, aprendizagem

ABSTRACT

The growth and great use of technology nowadays makes the courses and colleges turn to the technological area for a great demand, since the job market has a great impulse in this area. The major problem that occurs in technology colleges is the high dropout rate of students early in the course. Many academics who enter computer courses find it difficult to assimilate the course in programming, and often are not able to assimilate, so opting for dropping out of the course. And to find the alternatives for this fact, this work presents a comparative study among some tools that propose a programming environment with the aim of improving the learning of the programming discipline in a more dynamic and competitive way, as well as helping students. A solution should be used in classrooms. As we have already used Robocode, CodeCombat, CodeCademy and Codingame, as the ports have the same program scheduling challenges. Through a comparison method, it was possible to analyze the tools and their elements of gamification. The data studies and comparisons the performance in the work, was possible in the resource the multiple items the data over the work, and also may be used to logic of program programming.

Key words: programming, gamification, learning

LISTA DE FIGURAS

Figura 1 - Visualização do ROBOCODE	23
Figura 2 - IDE do ROBOCODE	24
Figura 3 - Interface do CodeCombat	27
Figura 4 - Sugestões do CodeCombat	27
Figura 5 - Linguagens do Codingame	28
Figura 6 - IDE do Codingame.....	29
Figura 7 - Linguagens do Codecademy	30
Figura 8 - Medalhas do Codecademy	30
Figura 9 – Características Especificas	32
Figura 10 - Comparativo sobre as Mecânicas de Jogo	34
Figura 11 - Comparativos sobre os Componentes de Jogo	35
Figura 12 - Total de Elementos Gamificados	36
Figura 13 - Elementos Gamificados para Educação	37

LISTA DE TABELAS

Tabela 1 - Mecânicas de Jogo	21
Tabela 2 - Componentes de Jogo	21
Tabela 3 - Elementos Gamificadas para Educação	22
Tabela 4 - Informações sobre o Codecombat.	26
Tabela 5 - Características Principais	31
Tabela 6 - Comparativos sobre Mecânicas de Jogo	33
Tabela 7 - Comparativos sobre os Componentes de Jogo	35
Tabela 8 - Elementos Gamificados para Educação	37

SUMÁRIO

1 INTRODUÇÃO	12
1.1 OBJETIVOS	13
1.1.1 OBJETIVO GERAL.....	13
1.1.2 OBJETIVOS ESPECÍFICOS	13
1.2 JUSTIFICATIVA	13
2 FUNDAMENTAÇÃO TEÓRICA	15
2.1 A APRENDIZAGEM EM LÓGICA DE PROGRAMAÇÃO	15
2.2 GAMIFICAÇÃO	16
2.2 SOFTWARE GAMIFICADOS UTILIZADOS PARA EDUCAÇÃO	16
3 PROCEDIMENTOS METODOLÓGICOS.....	19
3.1 A ESCOLHA DAS FERRAMENTAS	19
3.2 DEFINIÇÃO DO MÉTODO DE COMPARAÇÃO	20
4 ANÁLISE COMPARATIVA DAS FERRAMENTAS	23
4.1 CARACTERÍSTICAS DE CADA FERRAMENTA.....	23
4.1.1 ROBOCODE.....	23
4.1.2 CODECOMBAT	25
4.1.3 CODINGAME	28
4.1.4 CODECADEMY	29
4.2 CARACTERÍSTICAS INDIVIDUAIS	31
4.3 COMPARAÇÃO DAS FERRAMENTAS.....	33
4.4 COMPARAÇÃO DAS FERRAMENTAS VISANDO UM AMBIENTE EDUCACIONAL.....	36
5 CONSIDERAÇÕES FINAIS	39
REFERÊNCIAS.....	41

1 INTRODUÇÃO

Com a modernidade, diversas áreas estão aderindo a tecnologia para resolução de problemas do cotidiano, sendo esta necessária para realizar-se as tarefas habituais (NETTO, 2015). Com isso, há um aumento de empresas que buscam profissionais qualificados na área de tecnologia da informação.

Mesmo com o mercado de trabalho necessitando de profissionais na área de programação, no momento há uma certa insuficiência destes no Brasil. E isto diz respeito ao alto índice de desistência dos alunos nos cursos de tecnologia da informação (RAPOSO E DANTAS, 2016). Tal informação pode ser confirmada por (FLORENZANO, 2018) e também pelo INEP, que no ano de 2012 fez um levantamento e verificou que os cursos da área de exatas possui um índice de evasão de 28%.

Este alto índice de evasão pode ser ocasionado por inúmeros motivos. Têm-se informações de que muitos alunos, ao ingressarem em um curso de tecnologia, desistem por dificuldades ou até mesmo por não conseguirem acompanhar a disciplina de programação (PRIETCH e PAZETO, 2010), deste modo criando um obstáculo para o discente. (DE LIMA JUNIOR, VIEIRA e VIEIRA, 2015).

Baseando-se nessas informações, este trabalho irá propor uma comparação entre elementos gamificados de algumas ferramentas que podem ser um diferencial no aprendizado da disciplina de lógica de programação,-que poderá engajar os alunos na busca deste conhecimento.

Com os elementos de gamificação presentes nestas ferramentas, os alunos poderão ter um maior engajamento quanto à aprendizagem, pois segundo os autores Da Silva e De Queiroz (2014), os jogos são:

“Antes de qualquer coisa, sistemas de aprendizagem, e que isso dá conta do sentido de engajamento e entretenimento que os jogadores experimentam.” (Da Silva e De Queiroz 2014).

Com isto, será feita uma análise comparativa de algumas ferramentas, com o intuito de apontar as principais características, semelhanças e diferenças entre elas, identificando quais delas podem auxiliar os discentes na aprendizagem de programação de uma melhor forma.

1.1 OBJETIVOS

Nesta seção são apresentados o objetivo geral e os objetivos específicos que levaram ao desenvolvimento deste trabalho.

1.1.1 OBJETIVO GERAL

Realizar uma comparação de ferramentas que utilizam conceitos de gamificação no âmbito educacional. O objetivo é determinar as ferramentas com as melhores características gamificadas, que podem auxiliar os discentes na aprendizagem da disciplina de programação.

1.1.2 OBJETIVOS ESPECÍFICOS

Para alcançar o objetivo geral, será necessário atingir alguns objetivos específicos:

- Pesquisar softwares gamificados que se enquadram na temática da pesquisa;
- Comparar os métodos de gamificação de cada software;
- Identificar as características das ferramentas;
- Identificar quais dos softwares podem auxiliar a aprendizagem dos discentes de melhor modo.

1.2 JUSTIFICATIVA

Com o grande nível de evasão dos alunos da área de tecnologia (RAPOSO E DANTAS, 2016), faz-se necessário identificar alguns dos motivos para tal fato, sendo que estes motivos tornam-se barreiras para o discente prosseguir com a graduação (DE LIMA JUNIOR, VIEIRA e VIEIRA, 2015).

Segundo Prietch e Pazeto (2010), uma dessas causas é a disciplina de programação, tendo em vista que muitos alunos ingressantes na área não conseguem entender os conceitos básicos desta matéria.

E a gamificação pode ser um viés para tentar mudar este quadro, pois através dela o discente irá ter um maior engajamento na busca por conceitos voltados à disciplina.

Diante desse cenário, este trabalho irá propor uma análise comparativa de ferramentas gamificadas, visando identificar quais delas auxiliariam de melhor forma na aprendizagem dos discentes na disciplina de programação.

Pois como Ferreira (2011) afirma:

Jogos computadorizados funcionam como efeito estimulador no desenvolvimento intelectual, permitem maior flexibilidade de raciocínio, desafiam nosso funcionamento executivo, ajudam a treinar e estimular o pensamento lógico, o planejamento estratégico, a solução de problemas, a tomada de decisões, o reconhecimento de erros, a enfrentar situações novas, a inibir reações habituais quando se mostram inadequadas para o momento e o raciocínio dedutivo. (FERREIRA, 2011)

Este trabalho também é importante, visto que as ferramentas comparadas são gamificadas, e através destas comparações serão identificadas quais delas possuem mais elementos gamificados e qual se aplicaria melhor a educação. E softwares deste tipo podem engajar os discentes a buscar mais conhecimentos e motivá-los quanto à disciplina de programação.

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo são apresentados alguns temas que contribuíram para o aprofundamento e embasamento dos conceitos necessários para o desenvolvimento desta pesquisa. O primeiro tema abordado é sobre a aprendizagem em lógica de programação, seguido da gamificação, e, por fim, serão apresentados alguns softwares que já foram utilizados na educação.

2.1 A APRENDIZAGEM EM LÓGICA DE PROGRAMAÇÃO

Um dos maiores desafios para os alunos que ingressam em um curso voltado à área tecnológica é a disciplina de programação, pois nela existem elevados níveis de insucesso.

De acordo com Branco Neto e Schuvartz (2007) e De Lima, Vieira e Vieira (2015), acadêmicos iniciantes na área tecnológica se sentem incapazes de solucionar problemas com o uso da programação devido as habilidades que esta disciplina exige.

E esta incapacidade também se deve pelo fato dos acadêmicos não possuírem conhecimento sobre área, o que proporciona muitas desistências e reprovações nos cursos.

Uma das disciplinas que possui altos níveis de reprovação é a da programação, que tem como um de seus objetivos o desenvolvimento do raciocínio lógico, a fim de que o aluno obtenha conhecimentos e competências necessárias para que futuramente ele venha desenvolver soluções para os problemas do cotidiano.

No entanto, as disciplinas que têm em sua ementa lógica de programação, apresentam-se de maneira complexa para o aprendizado da maioria dos alunos, que possuem dificuldades para compreensão, até mesmo de conceitos simples como as estruturas de controle e estruturas de repetição. Essas dificuldades traduzem-se inevitavelmente em elevadas taxas de insucesso e desistência (LEA ET AL, 2003; KULYK ET AL, 2007) e (NETO, 2013)

Segundo Hundhausen et al. (2002), é necessário que alunos iniciantes consigam desenvolver habilidades para aplicar os conhecimentos adquiridos de linguagens de programação para a resolução de problemas.

2.2 GAMIFICAÇÃO

A atividade de gamificação teve início há muito tempo atrás. Um bom exemplo disso, é o ato de uma professora pontuar o aluno que respondeu corretamente uma questão (FADEL, *et al.*, 2014).

A gamificação foi usada pela primeira vez em 2002, e o responsável pelo ato foi Nick Pelling. No entanto, este termo só ganhou popularidade após oito anos, a partir de uma apresentação realizada por Jane McGonigal. Ela é uma famosa game designer norte-americana e autora do livro 'A realidade em jogo: Por que os games nos tornam melhores e como eles podem mudar o mundo?'. (VIANNA *et al.*, 2013).

“Gamificação é um processo que usa elementos, mecânicas e dinâmicas de jogos para engajar e direcionar comportamentos de jogadores dentro de um contexto de não jogo. (JUNIOR, 2014).”

Zichermann e Cunningham (2011), afirmam que as pessoas jogam por quatro razões específicas, tanto possuírem o domínio sobre determinado assunto, quanto o para alívio do stress, para entretenimento e socialização.

O uso da gamificação, segundo Fardo (2013), pode propor de forma significativa novos conhecimentos que proporcionam positivamente a experiência educacional dos envolvidos.-E através dela que os indivíduos terão um contexto para a construção de conhecimento mais amplo e interativo, tanto nas escolas como em outros ambientes acadêmicos.

Em resumo, o uso da gamificação em diferentes ambientes (locais de trabalho, escolas, universidade e entre outros) traz grandes vantagens. E isto se justifica porque uma vez que o paradigma do ambiente onde foi inserido a gamificação sofre uma alteração, as pessoas envolvidas começam a pensar de forma diferente, executando tarefas de forma espontânea e ágil.

2.2 SOFTWARE GAMIFICADOS UTILIZADOS PARA EDUCAÇÃO

Diversos estudos já foram realizados aplicando a gamificação em ambientes de ensino, não somente na área tecnológica, mas em diferentes áreas.

Os estudos apresentados nesta seção, obtiveram bons resultados quanto ao uso da gamificação no educação, sendo que, a seguir, tais pesquisas, trabalhos e seus resultados, aplicados na área de tecnologia, serão analisados.

Com o objetivo de tornar mais dinâmico o processo de ensino-aprendizagem da disciplina de Desenvolvimento de Aplicativos com Banco de Dados, do Curso Técnico em Informática da Escola Técnica Redentorista (Campina Grande – PB), Aguiar (2015) propôs um ambiente gamificado, criando metas para seus alunos por meio de exercícios.

Este trabalho, tinha como objetivo recompensar os acadêmicos para cada resolução correta, e com essa experiência o autor teve como resultado uma sala de aula mais participativa nas tarefas, além de ter notado um maior engajamento do alunos.

Por sua vez, Raposo e Dantas (2016) criaram um jogo para incentivar os alunos da disciplina da Introdução à Programação dos cursos de Bacharelado em Sistemas de Informação e Licenciatura em Ciências da Computação no campus IV da UFPB, localizada na cidade de Rio Tinto, a terem uma rotina de estudos fora da faculdade.

O jogo consistia em uma carga diária de exercícios selecionados pelo professor, e a resolução dos mesmos ocasionava em premiações que ajudariam no contexto do jogo. Como resultado desta pesquisa teve-se uma notável mudança no comportamento dos alunos em relação à rotina de estudos, bem como na motivação dos mesmos em relação à disciplina.

Outros autores que também aplicaram a gamificação com alunos foram Silva, Oliveira e Martins (2016). Eles criaram uma integração do jogo Minecraft com a linguagem de programação Python, por meio da API “Minecraft Python”. Este trabalho teve o mesmo intuito dos anteriores, que é motivar os alunos a programarem de forma dinâmica.

Um diferencial do trabalho de Silva, Oliveira e Martins (2016) foi a implementação de uma história junto à gamificação. Os resultados foram satisfatórios, pelo fato do método abordado ter proporcionado um aumento na motivação dos alunos envolvidos.

Contudo, houve uma dificuldade com o uso da API, de forma a evidenciar que ferramentas mais lúdicas e interativas possam ter um melhor desempenho.

Já Butt (2016), utilizou o Codingame com os alunos da Southampton Solent University, e concluiu que os acadêmicos são receptivos a tal abordagem. No entanto, observou que os alunos não se desfazem dos métodos já utilizados, e que uma mescla destes tipos de didáticas seria um bom método para aprendizagem.

Os trabalhos apresentados sobre gamificação obtiveram bons resultados quando aplicados em um ambiente de estudo, pois por meio dela foi capaz de promover um melhor engajamento dos alunos em relação ao conteúdo ministrado nas salas de aula.

3 PROCEDIMENTOS METODOLÓGICOS

Neste capítulo são apresentados os procedimentos metodológicos deste trabalho, e com isso é fornecido um aporte para o entendimento das etapas que foram realizadas para a análise das ferramentas, bem como a contextualização da pesquisa.

E, para que o objetivo geral fosse alcançado, foram realizados alguns procedimentos estruturados e definidos, tais como:

- Escolha das ferramentas a serem comparadas, bem como o motivo para esta escolha;
- Um estudo de cada ferramenta de forma individual, com o intuito de levantar suas principais características;
- Definição do método utilizado para a comparação das ferramentas.

Estas etapas foram fundamentais para alcançar os resultados almejados do objetivo geral, e serão contextualizadas nas seções seguintes, onde é descrito o motivo da escolha das ferramentas, bem como o método de comparação utilizado.

3.1 A ESCOLHA DAS FERRAMENTAS

Para a seleção das ferramentas que foram utilizadas nesta pesquisa, foram analisados alguns fatores para elencar algumas semelhanças entre elas, a fim de que a análise fosse realizada de forma igual sobre cada uma delas.

Primeiramente, foi analisado a disponibilidade das ferramentas quanto ao custo para sua aplicação, neste caso todas as ferramentas selecionadas são gratuitas.

Outro ponto importante que foi levado em consideração para a escolha destas ferramentas, foram as linguagens de programação oferecidas por cada uma delas, neste contexto todas as ferramentas possuem ao menos uma linguagem em comum.

Dentro deste cenário, as ferramentas selecionadas para a realização dentro trabalho foram:

- Robocode;

- Codingame
- CodeCombat
- CodeCademy

Na seção 4.1 deste trabalho, serão descritas as características individuais de cada ferramenta.

3.2 DEFINIÇÃO DO MÉTODO DE COMPARAÇÃO

Como método de comparação, foi utilizado os elementos de jogos denominados por Werbach e Hunter (2012), sendo estes também citados por Costa e Marchiori (2015), no qual analisaram estes elementos em conjunto com as áreas de aplicação da gamificação.

Este método foi utilizado pelo motivo de que estes elementos são citados em diversos trabalhos, tanto por Werbach e Hunter (2012) e Costa e Marchiori (2015).

Estes elementos de jogos identificados por Werbach e Hunter (2012), são categorias aplicáveis aos estudos e desenvolvimento da gamificação, e são divididos em três (dinâmicas, mecânicas e componentes), sendo estas categorias responsáveis por determinar o nível de gamificação de uma aplicação.

Contudo, neste trabalho foram utilizadas apenas duas categorias para a análise comparativa, sendo elas as Mecânicas e os Componentes de jogos. Tendo em conta que as dinâmicas, por se tratarem das emoções produzidas pelo jogo, tornam-se difíceis de mensurar na comparação.

Primeiramente, serão demonstrados as categorias de jogo (Mecânicas e Componentes), bem como os elementos presentes em cada uma, e em seguinte será realizada a análise sobre as ferramentas selecionadas.

Para Costa e Marchiori (2015), as mecânicas delimitam os jogadores dentro dos jogos, de forma a tratar o que eles podem ou não fazer dentro do jogo. Os elementos da mecânica estão descritos na Tabela 1.

“As mecânicas viabilizam o funcionamento do jogo e orientam as ações do jogador; e, dependendo da mecânica utilizada, os jogos podem ter uma variedade ampla de estilos (COSTA; MARCHIORI, 2015)”

Mecânicas	Descrição
Aquisição de Recursos	O jogador pode coletar itens que o ajudam a atingir os objetivos.
Avaliação (Feedback)	A avaliação permite que os jogadores vejam como estão progredindo no jogo.
Chance	Os resultados de ação do jogador são aleatórios para criar uma sensação de surpresa e incerteza.
Cooperação e Competição	Cria-se um sentimento de vitória e derrota.
Desafios	Os objetivos que o jogo define para o jogador.
Recompensas	O benefício que o jogador pode ganhar a partir de uma conquista no jogo.
Transações	Significa compra, venda ou troca de algo com outros jogadores no jogo.
Turnos	Cada jogador no jogo tem seu próprio tempo e oportunidade para jogar. Jogos tradicionais, como jogos de cartas e jogos de tabuleiros muitas vezes dependem de turnos para manter o equilíbrio no jogo, enquanto muitos jogos de computador modernos trabalham em tempo real.
Vitória	O "estado" que define ganhar o jogo.

Tabela 1 - Mecânicas de Jogo (FONTE: COSTA e MARCHIORI, 2015)

Já os componentes de jogos para Costa e Marchiori (2015), são aplicações visualizadas na interface do jogo, e são por meio deste que o usuário se interage com a aplicação. Os elementos dos componentes estão descritos na Tabela 2.

Componentes	Descrição
Avatar	Representação visual do personagem do jogador.
Bens Virtuais	Itens dentro do jogo que os jogadores podem coletar e usar de forma virtual e não real, mas que ainda tem valor para o jogador. Os jogadores podem pagar pelos itens ou moedas do jogo ou com dinheiro real.
Boss	Um desafio geralmente difícil no final de um nível que deve ser derrotado, a fim de avançar no jogo.
Coleções	Formadas por itens acumuladas dentro do jogo. Emblemas e Medalhas são frequentemente parte de coleções.
Combate	Disputa que ocorre para que o jogador derrote oponentes em uma luta.
Conquistas	Recompensa que o jogador recebe por fazer um conjunto de atividades específicas.
Conteúdos Desbloqueáveis	A possibilidade de desbloquear e acessar certos conteúdos no jogo se os pré-requisitos forem preenchidos. O jogador precisa fazer algo específico para ser capaz de desbloquear o conteúdo.
Emblemas/Medalhas	Representação visual de realizações dentro do jogo.
Gráfico social	Capacidade de ver amigos que também estão no jogo e ser capaz de interagir com eles. Um gráfico social torna o jogo uma extensão de sua experiência de rede social.
Missão	Similar a "conquistas". É uma noção de jogo de que o jogador deve fazer para executar algumas atividades que são especificamente definidas dentro da estrutura do jogo.
Níveis	Representação numérica de evolução do jogador. O nível do jogador aumenta à medida que o jogador se torna melhor no jogo.
Pontos	Ações no jogo que atribuem pontos. São muitas vezes ligadas a níveis.
Presentes	A possibilidade distribuir ao jogador coisas como itens ou moeda virtual para outros jogadores.
Ranking	Lista jogadores que apresentam as maiores pontuações/conquistas/itens em um jogo.
Times	Possibilidade de jogar com outras pessoas com mesmo objetivo.

Tabela 2 - Componentes de Jogo (FONTE: COSTA e MARCHIORI, 2015)

Sendo assim, as comparações realizadas neste trabalho foram para identificar a existência dos elementos gamificados de cada categoria proposta por Werbach e Hunter (2012) nas ferramentas selecionadas.

E também foram analisadas a existência dos elementos de gamificação necessários para a educação, de acordo com os resultados obtidos por Peixoto e Silva (2015), os quais realizaram uma pesquisa para identificar os elementos gamificados, que são essenciais para um software desta natureza ser aplicado à educação.

Esta pesquisa consistiu em analisar diversos artigos voltados à esse assunto, e com isso elencar os elementos mais citados em cada um deles.

Foram diversos elementos encontrados, porém os utilizados neste trabalho fazem parte do DMGL (Modelo Dinâmico para Gamificação na Aprendizagem), proposto por Kim e Lee (2015), e são aspectos funcionais do jogo, os elementos são os presentes na Tabela 3.

Elementos gamificados para Educação
Emblemas
Sistemas de Recompensas
Níveis
Placar do Líder
Metas
Desafios
FeedBack
Controle de Tempo
Pontos
Conexão com outras pessoas
Desbloqueio progressivo
Sistema Baseado em Tempo
Itens Virtuais

Tabela 3 - Elementos Gamificadas para Educação, segundo Peixoto e Silva (2015) (FONTE: O Autor)

As comparações sobre os elementos de Peixoto e Silva (2015), consistiu em analisar as ferramentas para verificar a existência de cada elemento proposto e, por fim verificar em quais delas há mais destes elementos.

4 ANÁLISE COMPARATIVA DAS FERRAMENTAS

Neste tópico, é realizada uma análise comparativa entre as ferramentas ROBOCODE, CODECOMBAT, CODINGAME, CODECADEMY, com o intuito de elencar importantes pontos que são o diferencial entre elas, e com isso apresentar de uma forma clara quais destas ferramentas podem ajudar, de uma melhor forma, os discentes na aprendizagem da disciplina de programação, de acordo com seus elementos gamificados.

4.1 CARACTERÍSTICAS DE CADA FERRAMENTA

Nesta seção, serão descritas as ferramentas e suas características, com o intuito de promover um embasamento do assunto para a comparação final.

4.1.1 ROBOCODE

O jogo Robocode foi originalmente iniciado por Mathew A. Nelson como esforço pessoal no final do 2000, e tornou-se profissional quando ele a trouxe para a IBM em julho de 2001 (Larsen, 2013).

Segundo Robocode Brasil, o Robocode é um jogo de programação, cujo o objetivo é desenvolver um robô para lutar contra robôs desenvolvidos por outros desenvolvedores. As linguagens de programação utilizadas para o desenvolvimento dos robôs são JAVA ou .NET.

A Figura 1 ilustra o funcionamento do jogo, na qual já foi iniciada uma batalha, sendo que nesta figura existem 5 jogadores independentes, onde já foram programadas as ações dos robôs.

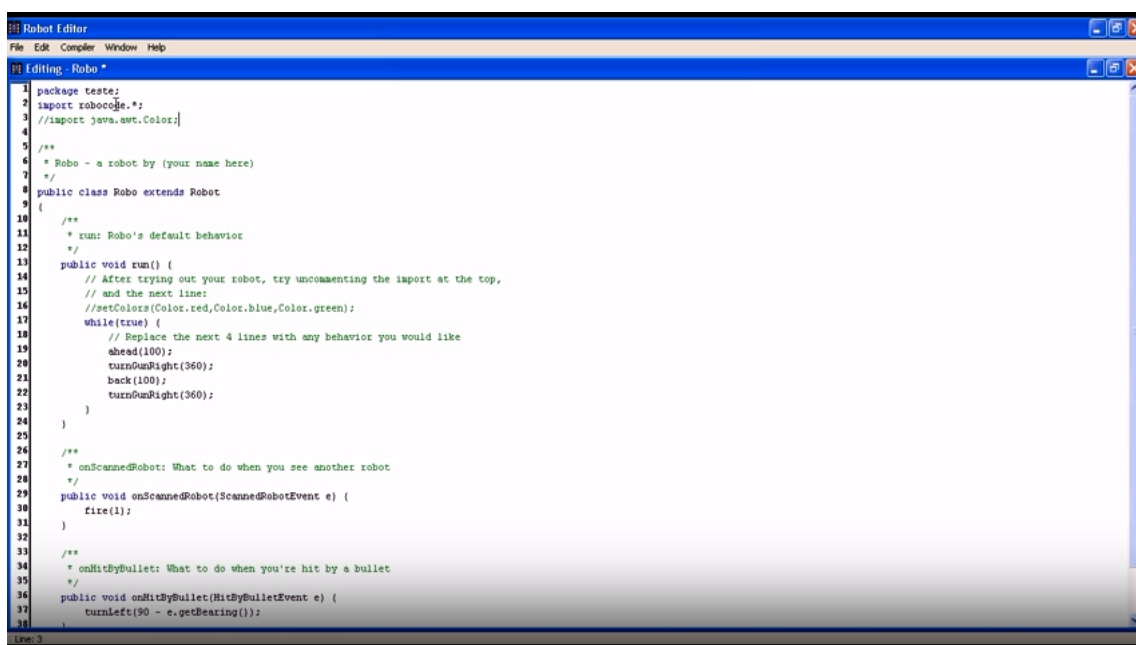


Figura 1 - Visualização do ROBOCODE

O Robocode conta com alguns modelos de robôs pré-programados, que possuem características específicas referentes ao método de funcionamento, e como irão se comportar a partir do momento em que a batalha for iniciada.

A ferramenta também possui métodos já desenvolvidos, sendo eles responsáveis pela performance do robô. Para utilizar este método o programador tem apenas a necessidade de instancia-los na programação.

O robôs podem ser programados por meio de uma extensão presente no software Eclipse, mas caso o usuário não queira, o Robocode também possui uma IDE de desenvolvimento própria, que está apresentada na Figura 2.

The image shows a screenshot of the RoboCode IDE. The window title is "Robo Editor". The menu bar includes "File", "Edit", "Compiler", "Window", and "Help". The main editor area shows a Java class named "Robo" that extends "Robot". The code includes package declarations, imports, and several methods: "run()", "onScannedRobot()", and "onHitByBullet()". The "run()" method contains a while loop with movement and rotation commands. The "onScannedRobot()" method calls "fire()". The "onHitByBullet()" method calls "turnLeft()".

```
1 package teste;
2 import robocode.*;
3 //import java.awt.Color;
4
5 /**
6  * Robo - a robot by (your name here)
7  */
8 public class Robo extends Robot
9 {
10     /**
11      * run: Robo's default behavior
12      */
13     public void run() {
14         // After trying out your robot, try uncommenting the import at the top,
15         // and the next line:
16         //setColors(Color.red,Color.blue,Color.green);
17         while(true) {
18             // Replace the next 4 lines with any behavior you would like
19             ahead(100);
20             turnGunRight(360);
21             back(100);
22             turnGunRight(360);
23         }
24     }
25
26     /**
27      * onScannedRobot: What to do when you see another robot
28      */
29     public void onScannedRobot(ScannedRobotEvent e) {
30         fire();
31     }
32
33     /**
34      * onHitByBullet: What to do when you're hit by a bullet
35      */
36     public void onHitByBullet(HitByBulletEvent e) {
37         turnLeft(90 - e.getBearing());
38     }
39 }
```

Figura 2 - IDE do ROBOCODE

Como mencionado, o Robocode é totalmente baseado no paradigma de orientação a objeto, e as linguagens utilizadas para o desenvolvimento dos robôs são JAVA e .NET.

Para o funcionamento do jogo, os desenvolvedores têm a função de “construir” seu tanque (ou também pode ser chamado de robô), com funções de ataque e defesa, pois como se trata de um combate, o ganhador será o último tanque sobrevivente após o fim da batalha.

Um fato importante a ser mencionado, é que depois da batalha ser iniciada o jogador só possuirá as funções programadas anteriormente, e não poderá fazer qualquer intervenção até o fim da batalha.

Com o funcionamento do jogo é possível notar alguns conceitos da orientação a objetos presentes no mesmo, tais como:

- Classes;
- Instâncias das Classes;
- Métodos;
- Envio de mensagens;

Segundo Júnior e Braga (2016), o jogo é composto por n rodadas (rounds), sendo n um parâmetro configurável, e essas rodadas se iniciam com os robôs com energia em nível 100. Quando o robô atingir o nível de energia 0, ele explode e por consequência sai da rodada atual do jogo.

Em relação aos gastos de energias, Braga (2016) explica que os tiros dos robôs possuem uma intensidade, sendo na faixa de 1 à 3, e o dano sofrido por um robô atingido será de quatro vezes este valor. Contudo, qualquer robô que efetuar um disparo, seja este de qualquer intensidade, sua energia também diminuirá proporcionalmente à intensidade do tiro efetuado, mas se este disparo efetuado atingir algum adversário, o robô receberá 3 vezes a intensidade do tiro em energia.

Ao se chocar com alguma coisa, o robô também perde pontos de energia, sendo 0.6 pontos ao se chocar com outro robô e 3 pontos ao se chocar contra as paredes.

Estas são algumas das características presentes na ferramenta do robocode.

4.1.2 CODECOMBAT

O CodeCombat é um jogo online que possui características de um ambiente medieval, e para a conclusão de seus níveis o jogador pode realizar a programação nas seguintes linguagens, Python (Padrão do Jogo), Javascript, CoffeScript e Lua, sendo que as duas últimas são linguagens experimentais.

O CodeCombat foi desenvolvido a partir de 2013, e hoje já possui mais de 5.000.000 (5 milhões) de usuários em todo o mundo, e estes estão espalhados em mais de 200 países. Com isso, ele já possui traduções para mais de 50 idiomas. (CODECOMBAT, 2018).

Além dos níveis e objetivos, este jogo tem funções que podem ser utilizadas para o ensino de lógica e linguagem de programação. Uma delas é a criação de salas e a separação de alguns desafios para seus alunos, bem como escolher a linguagem a ser trabalhada e as dificuldades referentes aos níveis.

O Codecombat possui 406 níveis que estão dispostos em 5 mundos mediáveis diferentes, cada um destes mundos aplica-se conceitos diferentes de programação. A Tabela 3 apresenta informações de cada mundo, bem como quantidade de níveis, tempo estimado de duração e conceitos abordados em cada um deles.

Mundo	Quantidade de Níveis	Tempo Estimado de Conclusão	Conceitos abordados
Masmorra Kithgard	39	1-3 horas	Sintaxe, métodos, parâmetros, strings, loops e variáveis
Floresta Backwoods	117	2-6 horas	If-else, operadores relacionados, propriedades do objeto e manipulação de entrada
Deserto Sarven	97	4-11 horas	Aritmética, contadores, while-loops, break, arrays, comparações de string, encontrando min/max
Montanha Cloudrip	109	N/D	Objetos literais, método de invocação remoto, for-loops, funções, desenhando, operação módulo
Geleira Kelvintaph	44	N/D	Técnicas avançadas

Tabela 4 - Informações sobre o Codecombat. (FONTE: CODECOMBAT, 2018)

O CodeCombat possui uma IDE própria como é visível na Figura 3. Os códigos serão escritos na parte direita, onde existe a opção de compilar para ver a animação que ocorre no canto esquerdo, local onde estão descritos os objetivos a serem cumpridos.

Estes objetivos são variados por fases, mas na maioria das vezes consiste em levar o personagem até os cristais, que são os objetivos de cada nível. Caso o usuário tenha dúvidas referentes aos métodos a serem utilizados, pode-se passar o mouse no campo "Methods", onde estão descritos quais são as funções necessárias para a realização do desafio proposto no nível.

Ao decorrer do jogo, o usuário vai adquirindo acessórios para o personagem, estes acessórios possuem a finalidade de deixar o personagem mais forte e resistente aos obstáculos presentes em cada nível.

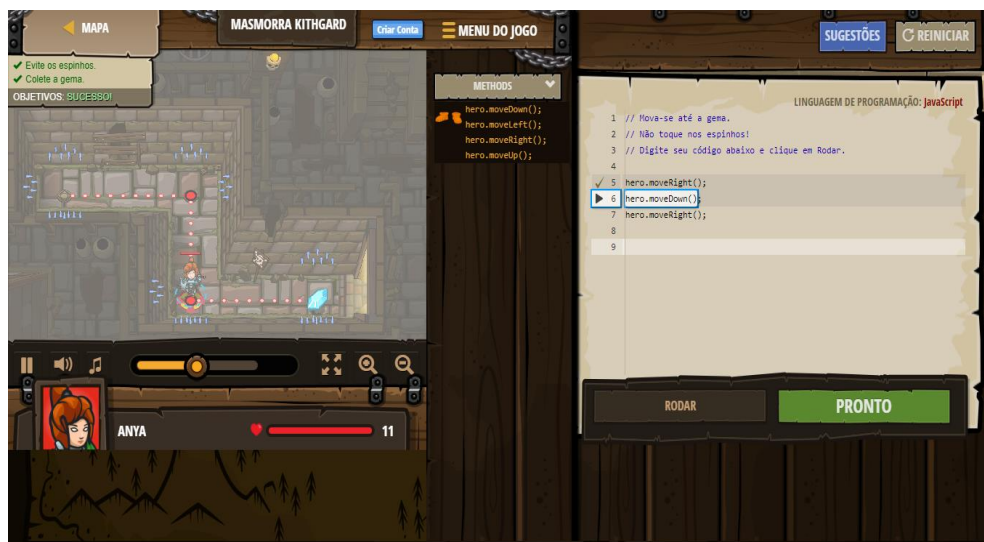


Figura 3 - Interface do CodeCombat

Caso o usuário encontre dificuldades para a resolução do nível, a ferramenta tem a opção para sugestões que mostrará diversas dicas referente ao código a ser escrito.

A Figura 4 ilustra a consulta, sendo que as setas azuis no quadro de sugestão indicam onde o personagem deve percorrer para alcançar o objetivo. Com base nestas setas o usuário tem a visão dos métodos que poderão ser utilizados.

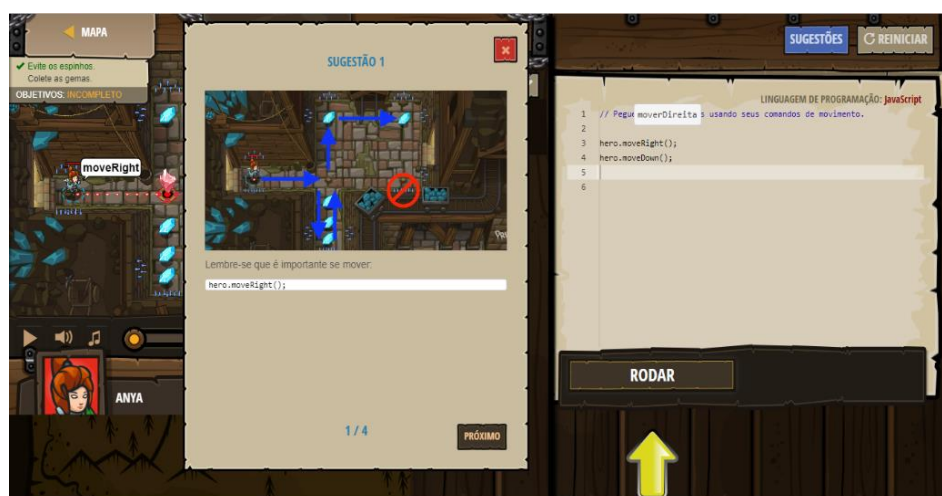


Figura 4 - Sugestões do CodeCombat

Outra característica do CodeCombat é o nível de dificuldade, que aumenta gradativamente. Este elemento é necessário para estimular a buscar por novos conceitos e conhecimentos. Com isso, a cada nível atingindo o usuário necessita utilizar mais funções para alcançar seu objetivo.

4.1.3 CODINGAME

Codingame também é uma plataforma de jogo online, gamificado e com isso o usuário pode competir em modo multiplayer, e a cada nível solucionado adquirir mais experiência, que no jogo é denominado como 'XP', podendo, de certa forma, abrir um ambiente de competição entre o mesmo e seus amigos.

Esta ferramenta possui suporte para mais de 25 linguagens de programação (Figura 5), e aborda todos os tópicos referente à elas (CODINGAME, 2018).



Figura 5 - Linguagens do Codingame (FONTE: CODINGAME, 2018)

Outro diferencial desta ferramenta é que, conforme o usuário vai progredindo nos níveis e ganhando mais experiência, seu perfil pode ser analisado por empresas que estão procurando profissionais para a área. Neste cenário o jogador pode selecionar quais as empresas que poderão ver o seu perfil dentro do jogo, podendo assim abrir oportunidades de emprego, por meio da plataforma do codingame.

Para acessar o CODINGAME é necessário criar um perfil que dará acesso ao ambiente do jogo e aos desafios que são de acordo com o nível de experiência do usuário, bem como mostrar um pouco do seu perfil (o nível em que o mesmo se encontra e quantas conquistas já obteve).

Após escolher o desafio, o usuário também deve escolher dentre as 25 uma linguagem de programação a qual será trabalhada.

Após a definição da linguagem, a IDE já estará disponível para a programação no canto direito da tela, sendo que ao lado esquerdo estará disposta a animação do jogo, e logo abaixo os requisitos do que deve ser feito para a resolução do problema proposto, Figura 6.

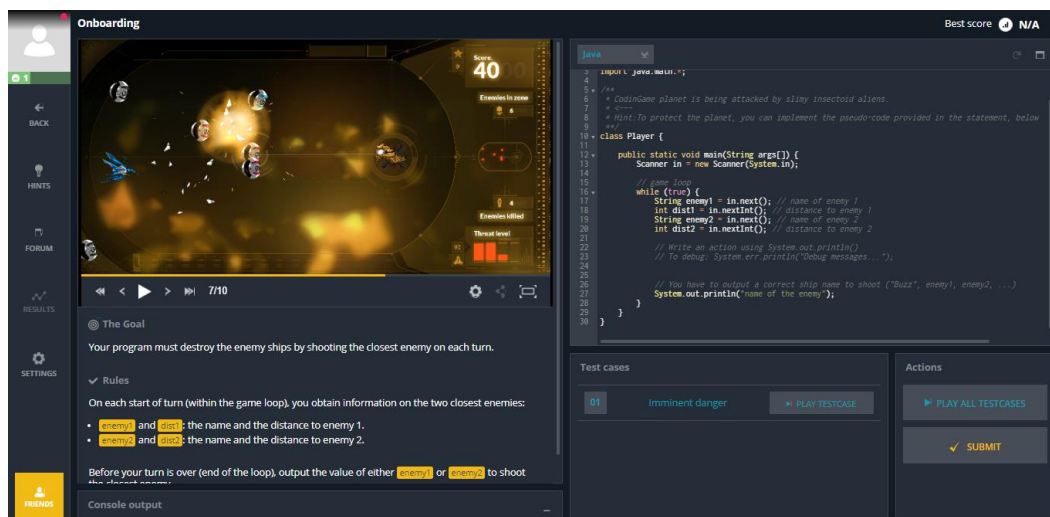


Figura 6 - IDE do Codingame

Depois de codificar a resolução do problema, o usuário deve submeter o código para que a ferramenta o leve para o próximo desafio.

4.1.4 CODECADEMY

O Codecademy é uma plataforma online em que há diversos cursos gratuitos para o usuário aprender a programar. Foi projetado por Zach Sims e Ryan Bubinski em 2011 (WORTHAM, 2011).

As linguagens de programação disponíveis na ferramenta são jQuery, Javascript, Python, Ruby, PHP, JAVA. (SUMMERS, 2013).

Ao realizar o cadastro, o usuário possui seu perfil individual, e como método de motivação a ferramenta oferece feedback e medalhas quando o usuário completa um objetivo. O jogador também possui a opção de criar seu próprio curso dentro da ferramenta. (CODECADEMY, 2018).

O Codecademy também conta com um fórum para que os usuários possam trocar experiências e conhecimentos entre si, (CODECADEMY, 2018)

Para mostrar um pouco da interface de uso do Codecademy, a Figura 7, apresenta as linguagens de programação disponíveis, na qual pode-se notar a progressão do usuário em cada uma delas.

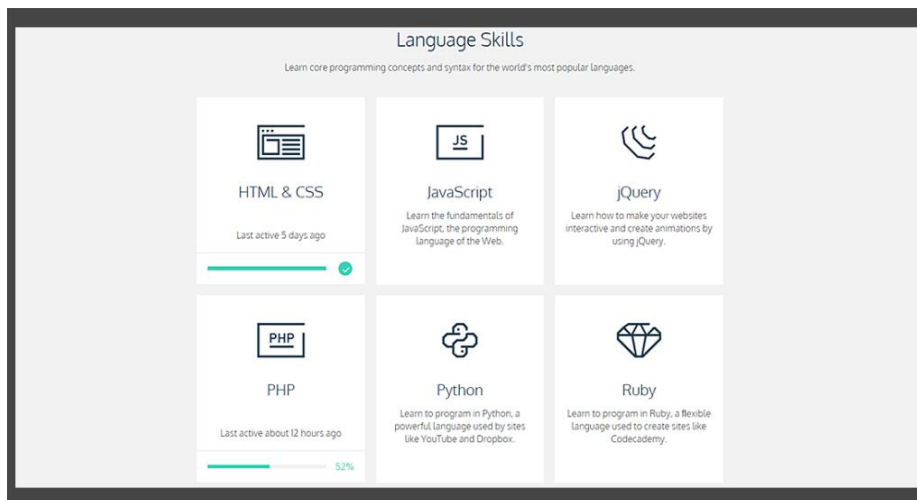


Figura 7 - Linguagens do Codecademy (FONTE: elearningsuperstars.com)

Como pode se notar na Figura 8, em que o usuário completou o curso de Ruby, nesta mesma tela fica disposto as medalhas adquiridas por meio da resolução dos itens.

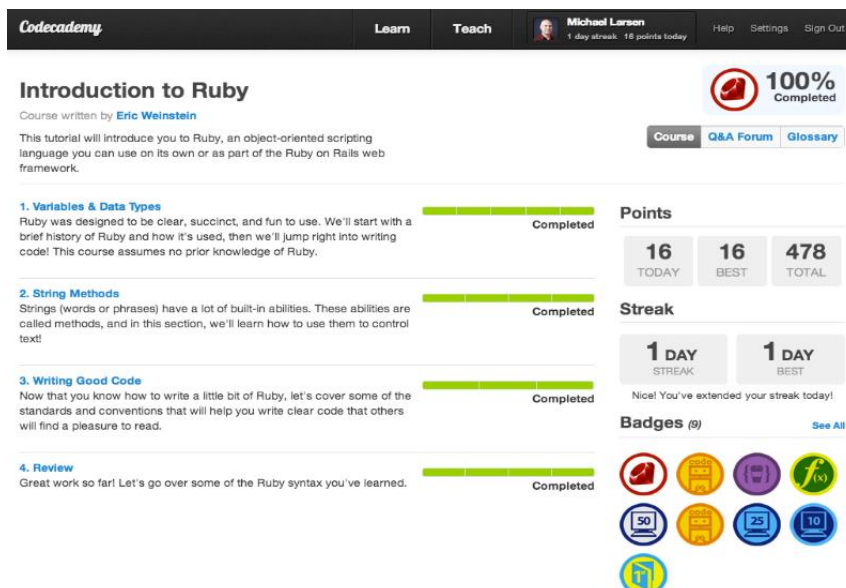


Figura 8 - Medalhas do Codecademy

Esta figura, também apresenta o tempo utilizado para conclusão do curso, bem como os pontos obtidos.

4.2 CARACTERÍSTICAS INDIVIDUAIS

Antes de começar a analisar as ferramentas e suas características, foi feito um levantamento, o qual está disposto na Tabela 4, que faz uma breve comparação entre elas, tal como público alvo ('M' – para Ensino Médio e 'S' – para Ensino Superior). As linguagens de programação suportadas por cada uma delas, o contexto no qual está inserido, cada ferramenta e também quais idiomas estão disponíveis para o público.

Ferramentas	Público Alvo	Linguagens	Contexto	Idioma
Robocode	M-S	JAVA .NET	Programação de Robôs para batalharem entre si.	Inglês
CodinGame	M-S	C, C++, C#, JAVA, JAVASCRIPT, PHP, PYTHON e outras 20 mais.	Desafios de programação variados.	Inglês e Francês
Codecombat	M-S	PYTHON JAVASCRIPT	Níveis a serem superados através de programação.	50 idiomas
CodeCademy	M-S	jQuery, JavaScript, Ruby, PHP, Python, JAVA	Cursos Online	Inglês, Espanhol, Francês e Português

Tabela 5 - Características Principais (FONTE: O Autor)

De acordo com as informações da Tabela 2, pode-se notar que todas as ferramentas possuem uma semelhança no que diz respeito ao uso das linguagens. A variedade em algumas delas surgem como um diferencial em relação às outras, sendo um ponto que vale-se ressaltar.

Em relação ao contexto de cada ferramenta, pode-se notar algumas diferenças, pois o Robocode, por exemplo, se aplica apenas em batalhas de robôs, já o CodeCombat e o CodinGame são jogos voltados à desafios e/ou níveis a serem superados, e por fim o CodeCademy possui um caráter mais sério, voltado à apenas lições sobre a programação.

Dentre as ferramentas analisadas, todas possuem suporte para vários idiomas, porém o Robocode e o Codingame não disponibilizam o idioma português.

Outro ponto que deve-se analisar, é que todas as ferramentas possuem um mesmo público alvo, para que a escolha delas não possa ter pesos por meio desse contexto.

Após analisar cada uma das ferramentas, foram identificados alguns pontos importantes sobre cada uma delas.

Primeiramente, sobre o método de funcionamento de cada ferramenta, foi analisado como é a execução de cada uma delas e o que deve ser feito antes do usuário começar a trabalhar com elas.

Dentre as quatro ferramentas comparadas, a única que não possui uma plataforma online é o Robocode, pois é necessário que seja instalado no computador do usuário, por outro lado não é necessário conexão com internet.

Já as ferramentas Codingame, CodeCombat e CodeCademy são plataformas online, e necessita-se apenas que o usuário crie um login pessoal para ter acesso à ferramenta. Há uma ressalva para o CodeCombat, que pode ser jogado de maneira anônima sem a necessidade de um cadastro.

A Figura 9 ilustra uma comparativo quanto a plataforma e ao ensino, aplicado por cada uma das ferramentas.

	Robocode	CodinGame	CodeCombat	CodeCademy
Quanto a plataforma?	Local - Sem necessidade de internet	Online	Online	Online
Quanto ao método de ensino?	Necessita-se de prévio conhecimento	Necessita-se de prévio conhecimento	Conteúdos em sequência	Conteúdos em sequência

Figura 9 – Características Específicas

Quanto ao ensino foi analisada a forma que o conteúdo de programação é apresentado ao jogador. Como pode-se analisar, as ferramentas Codecombat e CodeCademy possuem uma apresentação de conteúdo de forma sequencial, o que facilita na aprendizagem do usuário.

Já as ferramentas Rocabode e Codingame necessitam de um conhecimento prévio sobre o assunto. Fato que não desqualifica as mesmas para a utilização no ensino.

4.3 COMPARAÇÃO DAS FERRAMENTAS

Com base nos elementos propostos por Werbach e Hunter (2012) e também analisados por Costa e Marchiori (2016), as ferramentas serão comparadas de forma a identificar a existência dos elementos gamificados em cada uma delas, e com isso ter um embasamento para analisar qual ferramenta poderá ajudar na aprendizagem de uma melhor forma, por meio dos elementos gamificados.

Os primeiros elementos a serem comparados são as Mecânicas, que são peças fundamentais para orientar ações dos jogadores (COSTA e MARCHIORI, 2015). Na educação, são importantes por engajar e conduzir as pessoas na solução de problemas e promover o processo de aprendizagem (FRANÇA, REATEGUI e COLLARES, 2014).

Na Tabela 5 está demonstrado a disponibilidade dos elementos de Mecânicas de jogo, presentes em cada uma das ferramentas selecionadas.

Por meio dessa comparação é possível identificar quais dos softwares possuem mais elementos gamificados, a partir das mecânicas de jogo.

Mecânicas	Robocode	CodinGame	CodeCombat	CodeCademy
Aquisição de recursos	Não	Não	Sim	Não
Avaliação (Feedback)	Não	Sim	Sim	Sim
Chance	Sim	Não	Não	Não
Cooperação e competição	Sim	Sim	Sim	Sim
Desafios	Sim	Sim	Sim	Sim
Recompensas	Não	Sim	Sim	Sim
Transações	Não	Não	Não	Não
Turnos	Não	Não	Não	Não
Vitória	Sim	Sim	Sim	Sim

Tabela 6 - Comparativos sobre Mecânicas de Jogo (FONTE: O Autor)

Por meio das informações presentes na Tabela 5, a Figura 10 apresenta os resultados referentes à disponibilidade dos elementos gamificados em cada uma das ferramentas, de forma a apresentar quais delas disponibilizam mais elementos de mecânicas de jogo.

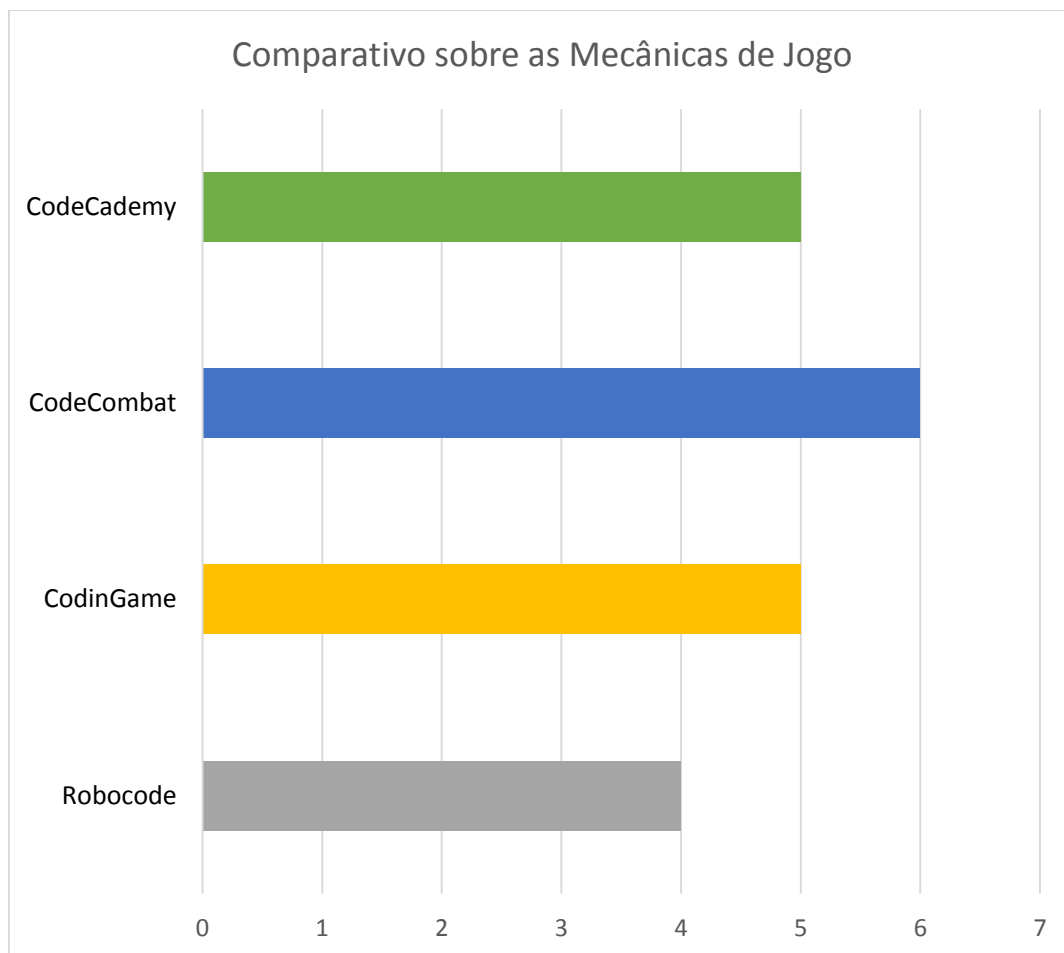


Figura 10 - Comparativo sobre as Mecânicas de Jogo

Como pode-se notar, a ferramenta que possui mais elementos gamificados voltados à mecânicas de jogo é o Codecombat, porém isso não demonstra que ele é melhor do que os demais, possuindo, apenas, mais elementos. Já o Robocode foi o que apresentou menos elementos neste quesito.

O próximo tópico avaliado é referente as disponibilidades de elementos de componentes de jogo presente em cada uma das ferramentas. A comparação segue o mesmo padrão da anterior, onde são elencados o que cada ferramenta disponibiliza à respeito de componentes. A comparação está apresentada na Tabela 6.

Componentes	Robocode	CodinGame	CodeCombat	CodeCademy
Avatar	Sim	Sim	Sim	Não
Bens virtuais	Não	Não	Sim	Não
Boss	Não	Não	Não	Não
Coleções	Não	Sim	Sim	Sim
Combate	Sim	Não	Não	Não
Conquistas	Não	Sim	Sim	Sim
Conteúdos desbloqueáveis	Não	Não	Sim	Não
Emblemas / Medalhas	Não	Sim	Sim	Sim
Gráfico Social	Não	Sim	Não	Sim
Missão	Não	Sim	Sim	Sim
Níveis	Não	Não	Sim	Não
Pontos	Não	Sim	Sim	Sim
Presentes	Não	Não	Não	Não
Ranking	Não	Sim	Não	Sim
Times	Não	Não	Não	Não

Tabela 7 - Comparativos sobre os Componentes de Jogo (FONTE: O Autor)

Por meio dos dados presentes na Tabela 6, é apresentado na Figura 11 um quantitativo dos elementos presentes em cada ferramenta, a fim de analisar quais delas provém de mais elementos gamificados referentes à componentes de jogo.

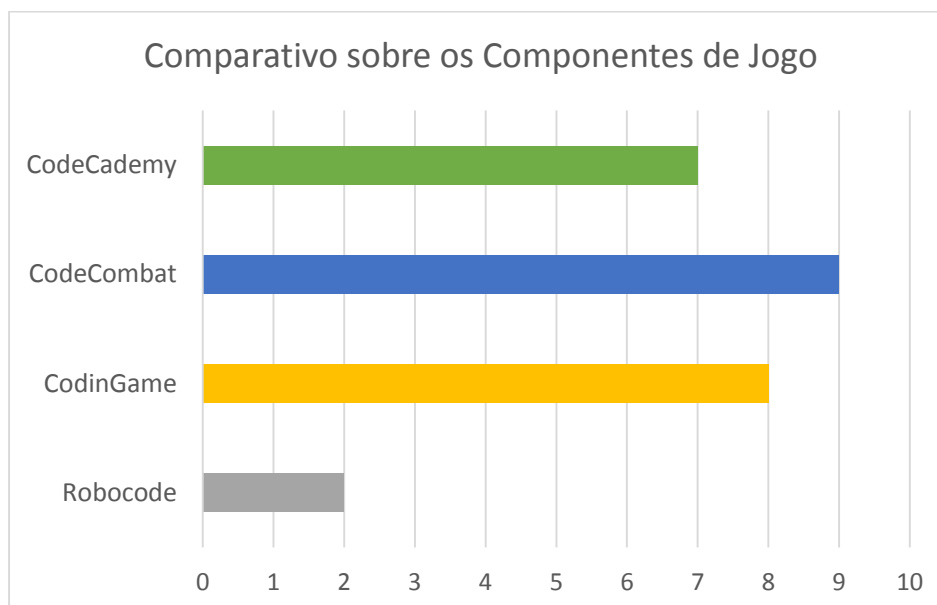


Figura 11 - Comparativos sobre os Componentes de Jogo

De acordo com os dados da Figura 12, é possível verificar que a ferramenta que mais possui elementos de componentes de jogos é o CodeCombat, e que o Robocode não possui muitos desses elementos.

Após realizar e demonstrar o levantamento de cada uma das ferramentas, em relação aos elementos de mecânicas e componentes de jogo, a Figura 13 apresenta um quantitativo geral das ferramentas sobre todos os elementos.

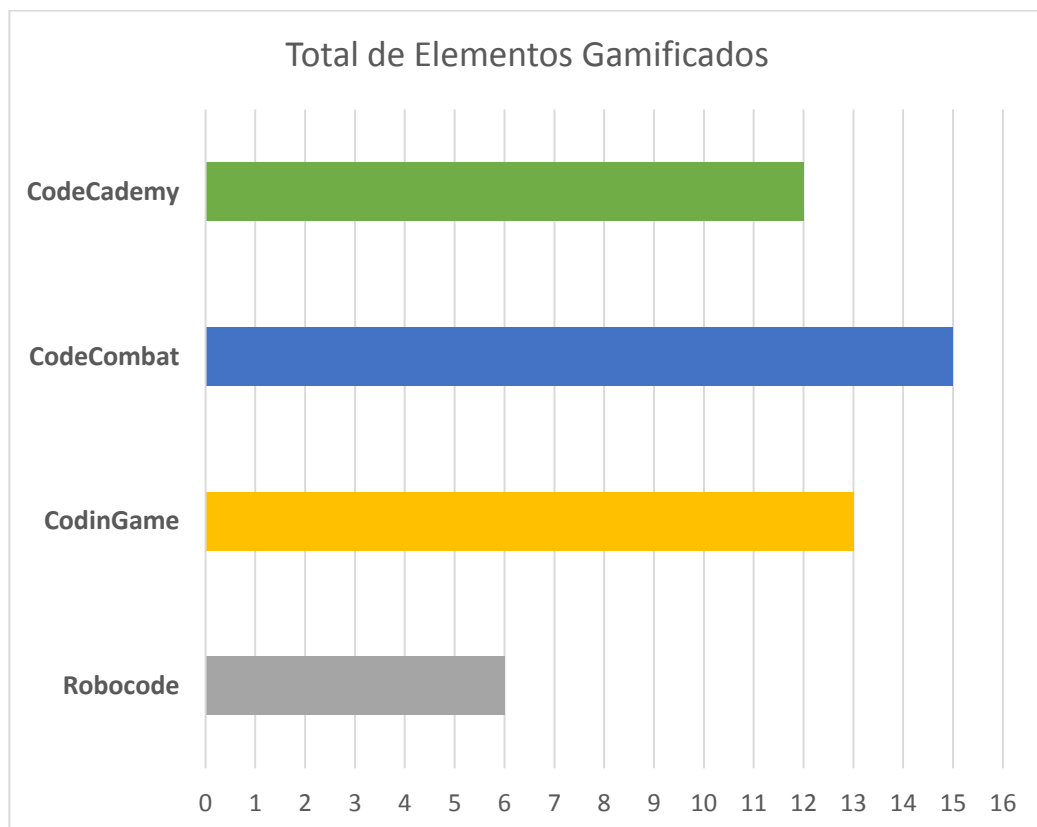


Figura 12 - Total de Elementos Gamificados

Por meio da Figura 13, é possível determinar que, dentre todas as ferramentas analisadas, a que possui mais elementos gamificados é o CodeCombat, seguido por CodinGame e CodeCademy, e sendo o Robocode a ferramenta que possui menos elementos gamificados.

4.4 COMPARAÇÃO DAS FERRAMENTAS VISANDO UM AMBIENTE EDUCACIONAL

Peixoto e Silva (2015) realizaram uma pesquisa para identificar na literatura quais os principais requisitos necessários para uma ferramenta gamificada ser aplicada na educação.

A Tabela 7 apresenta os elementos com maior índice de citação na literatura, segundo Peixoto e Silva (2015), e por meio destes elementos serão comparadas as ferramentas de forma a verificar a existência dos mesmos em cada uma.

Elementos gamificados	Robocode	CodinGame	CodeCombat	CodeCademy
Emblemas	Não	Sim	Sim	Sim
Sistemas de Recompensas	Não	Sim	Sim	Não
Níveis	Não	Não	Sim	Não
Placar do Líder	Não	Sim	Não	Sim
Metas	Sim	Sim	Sim	Sim
Desafios	Não	Não	Não	Não
FeedBack	Não	Sim	Sim	Sim
Controle de Tempo	Não	Sim	Não	Não
Pontos	Não	Sim	Sim	Sim
Conexão com outras pessoas	Sim	Sim	Não	Sim
Desbloqueio progressivo	Não	Não	Sim	Não
Sistema Baseado em Tempo	Não	Sim	Não	Não
Itens Virtuais	Não	Sim	Sim	Sim

Tabela 8 - Elementos Gamificados para Educação (FONTE: O Autor)

Cada elemento foi analisado nas ferramentas gamificadas de forma a verificar a existência do item. O gráfico da Figura 13, ilustra quantitativamente este resultado.

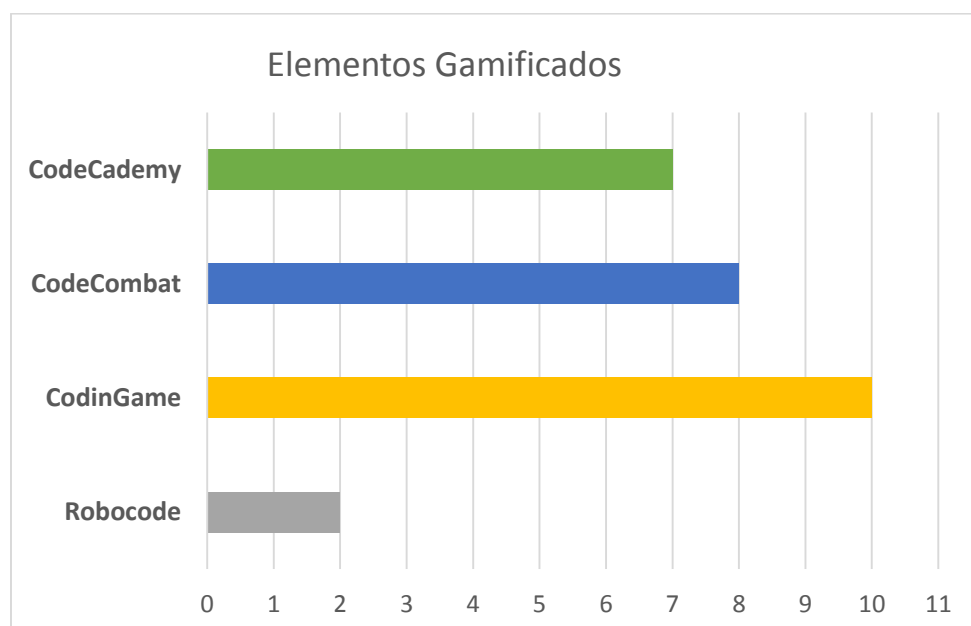


Figura 13 - Elementos Gamificados para Educação

De acordo com a pesquisa de Peixoto e Silva (2015), a ferramenta que possui mais requisitos para a aplicação na educação é o Codingame, seguido por CodeCombat e CodeCademy, já o Robocode se mostrou insuficiente quanto aos requisitos necessários.

Através dos resultados obtidos nas análises dos tópicos anteriores, pode-se notar que a ferramenta que mais se destacou por meio das categorias propostas por Werbach e Hunter (2012) é o CodeCombat, pois foi a ferramenta na qual se identificou uma maior quantidade de elementos de gamificação.

Porém, este destaque não se manteve por meio dos elementos voltados para a educação proposto por Peixoto e Silva (2015), na qual a ferramenta que mais possui tais elementos é o Codingame.

5 CONSIDERAÇÕES FINAIS

Este trabalho teve como objetivo realizar uma análise comparativa de ferramentas gamificadas voltadas a aprendizagem de programação, com o intuito de indicar qual das ferramentas comparadas seria a mais indicada para o uso acadêmico.

Esta trabalho foi desenvolvido visando encontrar alternativas para diminuir a evasão dos alunos nos cursos da área tecnológica, promovendo por meio da gamificação um ambiente de aprendizagem mais interativo e engajador.

Com definição das ferramentas gamificadas, foram descritas suas características principais, e em seguida foram analisados elementos de gamificação presentes em cada uma delas.

Os elementos analisados foram referentes as mecânicas e os componentes de jogos, e como resultado pode-se analisar que dentre as ferramentas analisadas, o CodeCombat foi o que mais se destacou, visto que dentre as ferramentas é o que mais possui tanto mecânicas quanto componentes de jogos.

E de acordo com as análises realizadas no que se diz respeito aos elementos gamificados para a educação, a ferramenta mais adequada para utilização no ensino seria o Codingame, na qual também se destacou por possui mais destes elementos.

Já o Robocode foi a ferramenta analisada que possui um menor número de elementos gamificados essenciais à educação, segundo os parâmetros de Peixoto e Silva (2015).

Considera-se que o objetivo geral foi alcançado, pois através do método utilizado para comparação pode-se identificar qual das ferramentas possui mais elementos de gamificação.

As comparações realizadas não definem qual a melhor ferramenta para a educação, pois foram verificados apenas os elementos de gamificação presente em cada uma delas.

Para trabalhos futuros e aprimoramentos dos resultados obtidos nesta pesquisa, sugere-se a aplicação destas ferramentas no ensino, para identificar de forma exploratória qual ferramenta obterá melhores resultados referente à

aprendizagem, podendo assim definir qual delas seria a ideal para a aplicação na educação.

REFERÊNCIAS

AGUIAR, J. J. B. Experiência baseada em Gamificação no Ensino sobre Herança em Programação Orientada a Objetos. **Congresso Brasileiro de Informática na Educação**, Campina Grande, 2015. 1444-1453.

BRANCO NETO, W. C.; SCHUVARTZ, A. A. Ferramenta Computacional de Apoio ao Processo de Ensino-Aprendizagem dos Fundamentos de Programação de Computadores. **Simpósio Brasileiro de Informática na Educação**, 2007.

BUTT, P. STUDENTS' PERCEPTIONS OF GAME-BASED LEARNING. **ICICTE**, 2017.

CODECADEMY. Codecademy. **Codecademy**, 2018. Disponível em: <<https://www.codecademy.com/>>. Acesso em: 24 junho 2018.

CODECOMBAT. O jogo mais envolvente para aprender programação. **CodeCombat**, 2018. Disponível em: <<https://br.codecombat.com/home>>. Acesso em: 28 Abril 2018.

CODECOMBAT. Sobre. **CodeCombat**, 2018. Disponível em: <<https://br.codecombat.com/about#story>>. Acesso em: 28 Abril 2018.

CODINGAME. Aproveite seu jogo de codificação. **Codingame**, 2018. Disponível em: <<https://www.codingame.com/start>>. Acesso em: 1 Maio 2018.

COSTA, A. C. S.; MARCHIORI, P. Z. Gamificação, elementos de jogos e estratégia: uma matriz de referência, Ribeirão Preto, 2015.

DA SILVA, L. R.; DE QUEIROZ, R. J. Aprendizagem baseada em jogos: Uma reflexão sobre o modelo de currículo da Quest to Learn. **Congresso Brasileiro de Informática na Educação**, Dourados, 2014. 86-90.

DAVID, H. Encapsulamento, Polimorfismo, Herança em Java. **Devmedia**. Disponível em: <<http://www.devmedia.com.br/encapsulamento-polimorfismo-heranca-em-java/12991>>. Acesso em: 5 Novembro 2016.

DE CARVALHO, V. A.; TEIXEIRA, G. F. **Programação orientada a objetos : Curso técnico de informática**. Instituto Federal Espírito Santo. Colatina, p. 134. 2012.

DE LIMA JUNIOR, J. A. T.; VIEIRA, C. E. C.; VIEIRA, P. D. 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**. Faculdade de Educação Tecnológica do Rio de Janeiro. Rio de Janeiro, p. 15. 2015.

ELEARNINGSUPERSTARS. CODECADEMY: SELF-GUIDED CODING COURSES. **elearningsuperstars**, 2015. Disponível em: <<http://www.elearningsuperstars.com/project/code-cademy/>>. Acesso em: 25 junho 2018.

FADEL, L. M. et al. **Gamificação na Educação**. São Paulo: Pimenta Cultural, 2014.

FARDO, M. L. **A GAMIFICAÇÃO APLICADA EM AMBIENTES DE APRENDIZAGEM**. Universidade de Caxias do Sul. Caxias do Sul. 2013.

FERREIRA, L. Jogos virtuais melhoram a atenção e a memória, segundo especialistas. **Uol**, 2011. Disponível em: <<http://noticias.uol.com.br/ciencia/ultimas-noticias/redacao/2011/02/08/jogos-virtuais-melhoram-a-atencao-e-a-memoria-segundo-especialistas.htm>>. Acesso em: 4 Novembro 2016.

FLORENZANO, C. Computação é o que mais sofre evasão em universidades públicas e privadas. **CBSI**, 2018. Disponível em: <<https://www.cbsi.net.br/2018/02/curso-de-computacao-e-um-dos-que-mais.html>>. Acesso em: 12 jul. 2018.

FRANÇA, R. M.; REATEGUI, E. B.; COLLARES, D. Mecânicas de Games em um Ambiente de Aprendizagem baseado em. **SBC – Proceedings of SBGames**, Rio Grande do Sul, 12 Novembro 2014. 476 - 483.

HUNDHAUSEN, C. D.; DOUGLAS, S. A.; STASKO, J. T. A Meta-Study of Algorithm Visualization Effectiveness. **Journal of Visual Languages & Computing**, 2002.

INEP. Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira. **INEP**, 2012. Disponível em: <<http://portal.inep.gov.br>>. Acesso em: 6 Novembro 2016.

JÚNIOR, Valdir Stumm; BRAGA, Paulo. Robocode Tutorial. Disponível em: <<http://stummjr.github.io/robocode/>>. Acesso em: 01 dez. 16.

JUNIOR, Silvio Alves dos Santos. Gamificação: introdução e conceitos básicos. Disponível em . Acesso em 01 de dez. 2016.

Kim, T. J., Lee, W. H. Dynamical model for gamification of learning (DMGL), Multimedia Tools and Applications, Springer US, (2013).

KULYK, O. et al. Human-centered visualization environments. **Lecture Notes in Computer Science**, p. 13-75, 2007.

LARSEN, F. N. ReadMe for Robocode. **Robocode**, 2013. Disponível em: <<http://robocode.sourceforge.net/docs/ReadMe.html>>. Acesso em: 16 Novembro 2016.

LEA, S. J.; STEPHENSON, D. Higher education students attitudes to student centred learning: beyond educational bulimia, *Studies in Higher Education*, n. 28, p. 321-334, 2003'.

NETTO, M. I. A importância da tecnologia na vida atual. **radioboanova**, 2015. Disponível em: <<http://radioboanova.com.br/importancia-da-tecnologia-na-vida-atual>>. Acesso em: 30 Maio 2018.

OLIVEIRA, F. C. S.; DA SILVA, J. A. L.; MARTINS, D. J. S. Storytelling e gamificação como estratégia de motivação no ensino de programação com Python e Minecraft. **SBC – Proceedings of SBGames**, 2017.

PEIXOTO, M. M.; SILVA, C. **Requisitos para Softwares Educacionais Gamificados**. Universidade Federal de Pernambuco. Recife, p. 14. 2015.

Prietch, S. S & Pazeto, T. A. (2010). “Estudo sobre a Evasão em um Curso de Licenciatura em Informática e Considerações para Melhorias”. Anais do VIII Workshop de Educação e Informática Bahia-Alagoas- Sergipe - WEIBASE 2010, Maceió/AL.

RAPOSO, E. H.; DANTAS, V. F. O Desafio da Serpente - Usando gamification para motivar alunos em uma disciplina introdutória de programação. **CONGRESSO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO**, Uberlândia, 2016. 577-586.

ROBOCODE. Robocode Brasil. **Robocode Brasil**, 2016. Disponível em: <<http://www.robocodebrasil.com.br/>>. Acesso em: 16 Novembro 2016.

SILVA, Jamille Anderson Luiz da; OLIVEIRA, Fábio Cristiano Souza; MARTINS, Danielle Juliana Silva. (2017) “Storytelling e gamificação como estratégia de motivação no ensino de programação com Python e Minecraft”. 2017. In: SBC – Proceedings of SBGames

SUMMERS, N. Codecademy: Hour of Code app for the iPhone lets you learn basic programming anytime, anywhere. **thenextweb**, 2013. Disponível em: <<https://thenextweb.com/apps/2013/12/09/codecademy-hour-code-app-iphone-teaches-basics-programming-anytime-anywhere/>>. Acesso em: 19 junho 2018.

VIANNA, Ysmar et al. Como reinventar empresas a partir de jogos. Rio de Janeiro: Mjv Press, 2013

WERBACH, K.; HUNTER, D. **For the win**: how game thinking can revolutionize your business. Philadelphia

WORTHAM, J. Codecademy Offers Free Coding Classes for Aspiring Entrepreneurs. **Bits**, 2011. Disponível em:

<https://bits.blogs.nytimes.com/2011/09/14/codecademy-offers-free-coding-classes-for-aspiring-entrepreneurs/?_r=0>. Acesso em: 20 junho 2018.

ZICHERMANN, Gabe; CUNNINGHAM, Christopher. Gamification by Design. Implementing Game Mechanics in Web and Mobile Apps. Canada: O'ReillyMedia, 2011.