UNIVERSIdade FEEVALE

Marcelo Warth de Moura

Uma Proposta de Aplicativo Gamificado

Como Ferramenta de Ensino

Novo Hamburgo

2017

Marcelo Warth de moura

Uma proposta de aplicativo gamificado

como ferramenta de ensino

Trabalho de Conclusão de Curso

apresentado como requisito parcial

à obtenção do grau de Bacharel em

Ciência da Computação pela

Universidade Feevale

Orientador: Débora Nice Ferrari Barbosa

Novo Hamburgo

2017

Agradecimentos

Gostaria de agradecer a todos os que, de alguma maneira, contribuíram para a realização desse trabalho de conclusão, em especial:

Meus pais que me apoiaram durante todo o processo e minha namorada, pela paciência que teve comigo durante todo este processo de conclusão de curso.

Resumo

Este trabalho tem como objetivo desenvolver um aplicativo gamificado que ensine lógica de programação e história da computação. Decidiu-se por este tema, pois muitos ingressantes dos cursos de computação não chegam muito longe no mesmo por não conseguirem desenvolver a lógica de programação necessária para progredir no curso. Com isto decidiu-se realizar o desenvolvimento deste aplicativo, onde foi utilizada a história da computação como motivador para a resolução dos desafios propostos. São apresentadas as vantagens da utilização de um aplicativo gamificado e como o mesmo foi desenvolvido. O trabalho foi realizado utilizando a metodologia Design Science Research, onde o foco é garantir que uma pesquisa seja reconhecida como sólida e relevante pela comunidade acadêmica e pela sociedade. O projeto foi aplicado em sala de aula nos componentes curriculares de Lógica e Algoritmos e no Projeto Jovem Aprendiz presente na universidade Feevale utilizando-se de pesquisa com o usuário e relatório observacional para obtenção dos resultados, verificando se é de interesse dos alunos a utilização de uma ferramenta como a que será apresentada em sala de aula.

Palavras-chave: História da Computação, Lógica de Programação, Gamificação, Aplicativo.

Abstract

This work aims to develop a gamified application that teaches programming logic and computational history. This subject was decided because many students of the courses of computation do not go very far in them for not being able to develop the programming logic necessary to progress in the course. With this it was decided to carry out the development of this application, where the history of computing was used as motivator to solve the proposed challenges. The advantages of using a gamified application and how it has been developed are presented. The work was done using the methodology Design Science Research, where the focus is to ensure that a research is recognized as solid and relevant by the academic community and society. The project was applied in the classroom in the curricular components of Logic and Algorithms and in the Young Apprentice Project present at Feevale University, using research with the user and observational report to obtain the results, verifying if it is in the students' interest to use a tool like the one that will be presented in the classroom.

Keywords: Computer history, Programming Logic, Gamification, App.

Lista de Figuras

Figura 3.1 – Interface Minecraft Hour of Code ...................................................................... 22

Figura 3.2 – Interface CodeCombat ........................................................................................ 22

Figura 4.1 – Tela Inicial .......................................................................................................... 28

Figura 4.2 – Tela de Fases ...................................................................................................... 29

Figura 4.3 – Tela Resolução de Fase ...................................................................................... 29

Figura 4.4 – Tela Resultados ................................................................................................... 30

Figura 4.5 – Comandos ........................................................................................................... 30

Figura 5.1 – Modelo Conceitual Simples ............................................................................... 37

Figura 5.2 – Caso de Uso ........................................................................................................ 38

Figura 5.3 – Diagrama de Sequência ...................................................................................... 40

Figura 6.1 – Tela seleção de fases ........................................................................................... 43

Figura 6.2 – História no Jornal ................................................................................................ 44

Figura 6.3 – História no monitor CRT .................................................................................... 44

Figura 6.4 – História no monitor LED/LCD ........................................................................... 45

Figura 6.5 – Tela de resolução de desafio ............................................................................... 46

Figura 6.6 – Tela de resultados ............................................................................................... 47

Figura 6.7 – Site ...................................................................................................................... 48

Figura 8.1 – Gráfico conhecimento sobre a história da computação disciplinas .................... 53

Figura 8.2 – Gráfico conhecimento sobre a história da computação Jovem Aprendiz ........... 54

Figura 8.3 – Gráfico aprendi sobre a história da computação disciplinas .............................. 54

Figura 8.4 – Gráfico aprendi sobre a história da computação Jovem Aprendiz ..................... 55

Figura 8.5 – Gráfico resolvo problemas de lógica facilmente disciplinas .............................. 56

Figura 8.6 – Gráfico resolvo problemas de lógica facilmente Jovem Aprendiz ..................... 56

Figura 8.7 – Gráfico considero que estou mais apto disciplinas ............................................. 57

Figura 8.8 – Gráfico considero que estou mais apto Jovem Aprendiz .................................... 57

Figura 8.9 – Gráfico fácil utilização do aplicativo disciplinas ................................................ 58

Figura 8.10 – Gráfico fácil utilização do aplicativo Jovem Aprendiz .................................... 58

Figura 8.11 – Gráfico informações claras disciplinas ............................................................ 59

Figura 8.12 – Gráfico informações claras Jovem Aprendiz ................................................... 59

Figura 8.13 – Gráfico auxilio para o meu ensino disciplinas ................................................. 60

Figura 8.14 – Gráfico auxilio para o meu ensino Jovem Aprendiz ........................................ 60

Figura 8.15 – Gráfico ferramenta como auxilio ao ensino disciplinas ................................... 61

Figura 8.16 – Gráfico ferramenta como auxilio ao ensino Jovem Aprendiz .......................... 61

Lista de Tabelas

Tabela 1.1 – Tabela DSR ........................................................................................................ 18

Tabela 5.1 – Selecionar História ............................................................................................. 38

Tabela 5.2 – Selecionar Década .............................................................................................. 38

Tabela 5.3 – Selecionar Desafios ............................................................................................ 38

Tabela 5.4 – Selecionar Desafios Customizados .................................................................... 39

Tabela 5.5 – Selecionar Número do Desafio .......................................................................... 39

Tabela 5.6 – Resolver Desafio ................................................................................................ 39

Tabela 5.7 – Carregar XML .................................................................................................... 39

Tabela 5.5 – Descrição diagrama de sequência ...................................................................... 41

Tabela 6.1 – Tabela comparação entre trabalhos relacionados e este projeto ........................ 49

Sumário

[Introdução 10](#_Toc486974174)

[1 Gamificação e motivação para aprender 14](#_Toc486974175)

[2 Metodologia 18](#_Toc486974176)

[3 A Proposta 21](#_Toc486974177)

[3.1 Estudo De Sistemas Para Apoiar A Proposta 21](#_Toc486974178)

[3.2 Proposta do aplicativo HistoCode 23](#_Toc486974179)

[3.3 Design 25](#_Toc486974180)

[3.4 História e Desafios 28](#_Toc486974181)

[4 Modelo conceitual 34](#_Toc486974182)

[4.1 UML 35](#_Toc486974183)

[4.1.1 Diagrama de Casos de Uso 35](#_Toc486974184)

[4.1.2 Diagrama de Sequência 37](#_Toc486974185)

[5 Desenvolvimento 39](#_Toc486974186)

[6 Percurso da Pesquisa 47](#_Toc486974187)

[7 Resultados da pesquisa 49](#_Toc486974188)

[8 CONCLUSÃO 60](#_Toc486974189)

[8.1 Contribuição do trabalho frente a trabalhos relacionados 61](#_Toc486974190)

[8.2 Trabalhos Futuros 65](#_Toc486974191)

[Apêndice 66](#_Toc486974192)

[Referências Bibliográficas 68](#_Toc486974193)

Introdução

“As tecnologias são ferramentas que possibilitam dinamizar e melhorar o ensino, utilizando uma metodologia mais personalizada, proporcionando atividades inovadoras, por meio de práticas criativas. [...] Há uma unanimidade em afirmar que é preciso ultrapassar as barreiras pedagógicas, inovar, deixar de ser o professor apenas “repassador” de conteúdos.” (SOMBRIO, HAEMING, ULBRICHT, 2014).

Hoje, as crianças desenvolvem-se manuseando tecnologia, fazendo com que tenham acesso ilimitado às informações. Dispositivos moveis podem ser de grande ajuda para o fazer pedagógico, tornando as aulas mais atraentes, estimulando a autonomia, curiosidade, criatividade e socialização entre os alunos (MOUSQUER, ROLIM, 2011), propondo com que o aluno possa explorar outras maneiras de aprender e o professor de ensinar.

Segundo Dalesio (2004 apud SQUIRE, 2005) nos últimos anos, os jogos foram marginalizados e depois passaram a ser bem vistos pela mídia, tecnologia e agora pelas indústrias educacionais. Educadores que utilizam ambientes virtuais passaram a aprender muito mais sobre como construir ambientes educacionais através dos jogos.

Prensky (2001) coloca que os estudantes de hoje estão cercados pela tecnologia, utilizando computadores, celulares, videogames e muitas outras ferramentas da era digital. Também coloca que os estudantes, em média, passam mais de 10 mil horas jogando videogames e somente 5 mil horas lendo.

Segundo o site Wikipédia[[1]](#footnote-1), um aplicativo é um software desenvolvido para ser instalado em dispositivos eletrônicos móveis, como smartphone, tablets e afins, abrangendo diversos conteúdos, sendo estes compras, jogos, gerenciamentos diversos entre outros.

Segundo o site Wikipédia[[2]](#footnote-2) e o dicionário Aurélio[[3]](#footnote-3), um jogo é um exercício de passatempo, que serve como divertimento, deve existir um jogador, ou mais de um, e regras sobre as quais o jogador deve cumprir, o jogo deve possuir um objetivo. Um jogo é praticado para fins recreativos e em alguns casos, como parte de instrumentos educacionais, onde são usados para passar uma mensagem/conteúdo.

Este projeto trata-se de um aplicativo gamificado. Segundo Vianna et al. (2013), gamificação corresponde ao uso de mecanismos de jogos voltados ao objetivo de resolver problemas práticos ou de despertar o envolvimento entre um público específico. Zichermann e Cunnigham (2011) identificam que o envolvimento de qualquer público deve estar baseado em estruturas de recompensa, reforço e feedbacks, suportadas por mecânicas e sistemáticas que potencializam o envolvimento do indivíduo. Os mecanismos encontrados em jogos funcionam como um motor motivacional do indivíduo, contribuindo para o engajamento deste nos mais variados aspectos e ambientes. Quando os conceitos de gamificação, ou seja, recompensa, reforço e feedback são aplicados nos aplicativos, tem-se um cenário motivador para aprendizagem.

Sombrio et al. (2014) destaca que os professores são somente “repassadores” de conteúdo. Isto se deve a um sistema de ensino que tem tido dificuldades em acompanhar a evolução tecnológica da sociedade. Squire (2005) começou a utilizar o jogo Civilization III como ferramenta de ensino como parte de sua pesquisa. A partir disso percebeu que os alunos que utilizavam o jogo para estudos eram mais motivados a buscar mais conteúdos sobre os assuntos abordados. Porém, como o próprio autor coloca, muitos duvidavam como este jogo poderia ser utilizado em sala de aula, incluindo os alunos, e muitos destes desistiram do conteúdo, mudando de currículo. Porém os que permaneceram e se motivaram perceberam que era uma ótima maneira de aprender geografia e história e aprimorar a linguagem.

Autores como Aguilera (2003), perceberam que através dos anos, jogadores possuíam habilidades mais aprimoradas em relação aos não jogadores, como:

* Percepção espacial e identificação de objetos;
* Desenvolvimento de discernimento visual e separação de atenção visual;
* Desenvolvimento de lógica intuitiva;
* Desenvolvimento cognitivo de aspectos científicos e técnicos;
* Desenvolvimento de habilidades complexas;
* Representação espacial.

Devido a pesquisas como estas, foi percebido que um jogo para ser atrativo, deve motivar, desafiar, incentivar a curiosidade, competição, possuir fantasia e controle sobre o assunto. Isto demonstra que um jogo não deve possuir somente um objetivo, mas deve propor elementos que estimulem a pessoa ao mesmo, que a permita se aprofundar nos elementos do jogo e apoderar-se do conhecimento necessário para avançar.

O ensino de lógica é algo que deve ser considerado na educação básica, pois como Valesco (2009 apud SOUZA, 2013) aponta, não há garantia que ensinar lógica seja o suficiente para que o pensamento do aluno se torne autônomo, mas contribui para pensar de forma organizada, auxiliando na habituação de conceitos lógicos básicos, ajudando a apoiar numa postura crítica e argumentativa.

Segundo Abar (2006 apud SCOLARI, 2007), “o aprendizado da lógica auxilia os estudantes no raciocínio, na compreensão de conceitos básicos, na verificação formal de programas e melhor os prepara para o entendimento do conteúdo de tópicos mais avançados”.

Segundo o IDEB 2015[[4]](#footnote-4) (Índice de Desenvolvimento da Educação Básica), que é calculado a partir da média de proficiência em língua portuguesa e matemática por instituição de ensino, a nível nacional, o índice de 2015 ficou abaixo (4.5) da projeção que havia sido feita (4.7). Isto indica que os alunos estão com dificuldades nestes conteúdos. Para compreensão linguística e matemática, é importante o desenvolvimento do raciocínio lógico. Em função desses elementos, pensou-se na ferramenta proposta, pois como já expondo acima, este tipo de aplicação pode ajudar no desenvolvimento do aluno, principalmente no desenvolvimento do raciocínio lógico, facilitando o aprendizado.

Já existem ferramentas para o ensino de lógica e programação, como o code.org[[5]](#footnote-5) e o Scratch[[6]](#footnote-6) que tem como objetivo ensinar programação desde a infância de forma facilitada. Este projeto utilizará como meio de aproximar o usuário do aplicativo à história da computação, pois é um aplicativo e não um jogo, focado em ensinar história da computação e lógica. Além disso, esta proposta propõe um conjunto menor de comandos, facilitando o desenvolvimento das atividades que envolvem lógica.

Como já dito acima, vislumbrou-se aí a possibilidade de criar elementos que dessem suporte para o ensino da lógica e da história da computação de forma móvel. Neste sentido, o problemática da pesquisa consiste em: Desenvolver um aplicativo gamificado que auxilie no desenvolvimento do raciocínio lógico e computacional utilizando como temática a história da computação.

Decidiu-se utilizar o ensino de lógica, pois como citado acima por Valesco (2009 apud SOUZA, 2013), a mesma contribui para o pensamento organizado e habituação de conceitos lógicos básicos, que são extremamente necessários na área da informática para o desenvolvimento de softwares. A história da computação será apresentada ao aluno para que o mesmo tenha certo conhecimento sobre como a tecnologia se desenvolveu até o presente momento e o motivo de estudarmos a mesma para o futuro.

O aplicativo será desenvolvido utilizando de elementos de gamificação, ou seja, utilizando características dos jogos no mesmo, como: recompensas, reforço, feedback e demonstração do desenvolvido.

Além disso, seria utilizado como parte do projeto de extensão - Ensinando lógica com as Tecnologias da Informação - Programa Ciências e Tecnologias na Educação. O objetivo do projeto consiste em desenvolver o raciocínio lógico junto aos alunos das séries finais do ensino fundamental, através da utilização de ferramentas e recursos provenientes da tecnologia da informação.

No entanto, devido a datas disponíveis para utilização no projeto serem inapropriadas para avaliação adequada dos resultados, decidiu-se aplicá-lo nos seguintes componentes curriculares, Lógica e Algoritmos, onde os alunos são ingressantes dos cursos Ciências da Computação e Sistemas de Informação podendo assim haver uma avaliação parecida com a proposta inicial. Também foi aplicado no Projeto Jovem Aprendiz, que foca na preparação do jovem para o mercado de trabalho e cidadania de jovens com famílias de baixa renda, tendo como base a formação tecnológica, assim podendo ser um foco de estudo deste projeto.

# Gamificação e motivação para aprender

Segundo o dicionário Aurélio[[7]](#footnote-7) e o site Wikipédia[[8]](#footnote-8), jogo é uma atividade cuja natureza ou finalidade é a diversão/entretenimento. Existem diversos tipos de jogos, de mesa, caneta e papel, cartas, dados, tabuleiro entre outros, mas nas últimas décadas, a seção que tem mais se destacado é a de jogos virtuais. Estes jogos utilizam-se do lúdico para motivar o usuário a permanecer e aprender com o mesmo. Os jogos digitais são divididos em diversas categorias, como: ação, aventura, cassino, lógicos, estratégicos, esportivos, *role-playing games* (RPGs), entre outros.

Cada uma das categorias citadas pode ajudar em áreas distintas de aprendizagem, segundo Tarouco (2004), como:

* Ação: pode desenvolver reflexo, coordenação olho-mão.
* RPG: interação entre usuários.
* Estratégicos: desenvolver a sabedoria e habilidades de negócio quanto à construção ou administração de algo.

Rapkiewicz (2006) aponta que a falta de motivação e alta taxa de reprovação na cadeira de algoritmos, pelo fato de o conteúdo exigir com que o aluno deva pensar, enquanto até este momento o aluno devia somente decorar o conteúdo. Em seguida exemplifica algumas características que fazem com que um jogo seja atrativo e o porquê de utilizar jogos como ferramenta de ensino e como pode ser útil para explicar conteúdos como o da cadeira de algoritmos. Ele propõe o uso de jogos na fase introdutória de ensino de algoritmos e programação, para que sejam utilizados recursos visuais para minimizar os problemas citados acima.

Jesus (2007) relata os problemas no ensino tradicional e como as instituições de ensino deveriam acompanhar a tecnologia para potencializar o processo de ensino e aprendizagem, diminuindo a taxa de reprovação nas cadeiras iniciais de cursos de computação. Exemplifica como um objeto de aprendizado (digital ou não-digital) pode ser utilizado como ferramenta para treinamento baseado no computador, ambientes interativos, ambientes de aprendizagem colaborativa, entre outros.

Scaico (2012) demonstra inicialmente uma preocupação de que cada indivíduo retém informações de maneira particular. Para que estas informações sejam retidas mais facilmente, elevando a qualidade do aprendizado, jogos eletrônicos podem ser utilizados, pois projetam muitos estímulos para o usuário. Afirma que jogos têm o potencial educativo gerando inclinação para aprender, devido ao usuário necessitar do aprendizado do conteúdo em questão para poder prosseguir no jogo.

Um dos elementos utilizados atualmente na aprendizagem é a gamificação. Segundo Fardo (2013), gamificação deriva diretamente da popularização dos jogos e de suas capacidades de motivar ações, resolver problemas e potencializar aprendizagem. Ou seja, utilizar-se de características encontradas normalmente nos jogos, como história, feedback, cooperação, competição, regras, entre outros. Através destas características obtem-se o mesmo grau de envolvimento de um jogo comercial de um software gamificado. A gamificação visa propiciar que o usuário permaneça interessado no aplicativo, fazendo com que queira progredir e aprender o conteúdo apresentado para melhorar seu desempenho nos próximos desafios propostos para ele.

Segundo Zichermann e Cunnigham (2011), a gamificação consiste na utilização de características de jogos, como pensamentos e mecânicas, como forma de envolver usuários na busca por algum objetivo ou resolução de algum problema. Quando os conceitos de gamificação são aplicados, tem-se um cenário motivador para aprendizagem.

Este cenário de gamificação é, segundo Muntean (2011), caracterizado por três atributos dos seguintes tipos:

* Mecânica de Jogos: Progressão, Retroalimentação, Comportamental.
* Benefícios: Lealdade, Engajamento, Tempo Gasto, Influência, Divertimento, Viralidade.
* Personalidade: Exploradores, Realizadores, Sociáveis e Matadores.

Primeiro é necessário identificar a personalidade de quem queremos atingir, depois se verifica quais tipos de conteúdo estarão sendo apresentados e quais mecânicas serão utilizadas, com isto, quais os objetivos que se espera atingir, assim caracterizando os benefícios da gamificação.

O ensino de lógica e algoritmos faz falta para o usuário, como coloca Oliveira et al. (2014). O ensino básico de computação (lógica e algoritmos) deve ser apresentado desde o início da vida escolar. É onde muitos dos alunos têm maior dificuldade, ou seja, idealizar e realizar um pensamento lógico e algorítmico sem influência externa de outros alunos e professores. Obtendo este conhecimento, mesmo que o aluno não siga profissionalmente na área da computação, ele terá desenvolvido competências de resolução de problemas que serão úteis em qualquer área de atuação.

Para entender a motivação do usuário neste trabalho, precisamos entender os tipos de motivação que existem, segundo a área da psicologia, Neves (2004) destaca que existem as motivações intrínseca e a extrínseca.

A motivação extrínseca, tem sido definida como uma motivação em que a pessoa trabalha, desenvolve e demonstra algo em relação a uma tarefa, tendo como objetivo a obtenção de recompensa externa, materiais ou sociais, com a finalidade de atender a solicitações ou pressões de outras pessoas, ou demonstrar capacidade ou habilidade.

Já a motivação intrínseca é caracterizada com uma tendência natural de obter novidades e desafios. A pessoa realiza uma tarefa por considerá-la interessante, lhe conceder satisfação ou ser atraente. Caracteriza-se pela motivação do usuário com relação ao assunto, gerando autonomia do mesmo e a autoaprendizagem. Este é o ponto em que o trabalho que será apresentado nos próximos capítulos se concentra, pois o usuário será motivado a utilizar o projeto como ferramenta de auto aprendizado e não para provar competência ou capacidade para outros.

Para a motivação do usuário, toda ferramenta de ensino e/ou jogo possui uma história. Esta história pode ser fictícia ou não, desde que cumpra o objetivo de motivar o usuário a continuar a utilizar o mesmo. Este projeto contará com a história da computação como meio motivacional, utilizando-a como ferramenta para ensinar sobre a mesma e criar os problemas de lógica.

Como citado anteriormente, um jogo é toda e qualquer atividade em que exista um jogador e regras, e um aplicativo é um software desenvolvido para ser instalado em dispositivos eletrônicos móveis que pode abranger diversos conteúdos.

Com isto, este trabalho será caracterizado como um aplicativo gamificado, pois poderá ser utilizado através de dispositivos móveis e possuirá regras e jogadores, como um jogo, possuindo uma interface interativa ao usuário, apresentará o conteúdo de forma gráfica, mostrando assim para o usuário o funcionamento de suas criações, apresentará resultados ao mesmo, estimulando-o a sempre obter o melhor resultado durante as etapas do aplicativo através de um sistema de estrelas, cada fase possuindo três delas, onde uma representa que a fase foi concluída, outra que o número de comandos foi igual ou menor a um valor estabelecido e outra que indica se o número de iterações para a conclusão foi menor ou igual a um valor estabelecido. O aplicativo estará integrando a história da computação durante as fases do aplicativo, fazendo com que o usuário seja apresentado novamente a história da computação também através da iteração com o aplicativo.

# Metodologia

Este projeto será desenvolvido através da técnica de pesquisa DSR (*Design Science Research*) método bastante utilizado nas áreas de Engenharia de Produção, Administração, Economia e Gestão de Operações. Esta metodologia tem como objetivo garantir que uma pesquisa seja reconhecida como sólida e relevante pela comunidade acadêmica e pela sociedade. Será utilizado o trabalho de Lacerda et al. (2013) como base de conhecimento sobre DSR, o mesmo chama-se "*Design Science Research*: método de pesquisa para engenharia de produção". Nele são compilados diversos autores que falam sobre o tema, explicando como funciona esta metodologia.

A DSR possui diversas etapas para a pesquisa, a primeira é a Conscientização, na qual é compreendido o problema que está se tentando resolver ou diminuir. A segunda é a Sugestão, nela são colocadas possíveis soluções de artefato (programa ou produto a ser desenvolvido e/ou já existente) para o problema proposto. A terceira é o Desenvolvimento, onde este artefato é criado para ser aplicado. A quarta é a Avaliação, na qual é verificado como ele será testado e como seus resultados serão medidos. A quinta e última etapa é a Conclusão, que é a parte onde o processo geral é formalizado e os resultados deste artefato são apresentados para a comunidade acadêmica.

Estas etapas então são compiladas na tabela abaixo, onde é explicado qual a relevância do problema, o artefato a ser desenvolvido, como foi buscada a solução, o rigor da pesquisa, a avaliação, as contribuições e a comunicação da pesquisa. Por ser uma tabela resumida sobre as informações deste projeto, ao final de cada etapa haverá uma indicação de onde mais informações podem ser obtidas dentro deste documento, podendo haver etapas que ainda estão em planejamento e não possuirão um capitulo neste presente momento do projeto.

Tabela 1.1 – Tabela DSR

|  |  |
| --- | --- |
| **Diretriz** | **Aplicação da Diretriz na Pesquisa** |
| Relevância do Problema | Problema: Baixo número de usuários que permanecem nos cursos de computação devido à baixa capacidade para resolver problemas lógicos e matemáticos durante os primeiros semestres do curso.  Relevância: Segundo o IDEB2015, que é calculado a partir da média de proficiência em língua portuguesa e matemática por instituição de ensino, a nível nacional, o índice de 2015 ficou abaixo da projeção que havia sido feita, isto indica que os alunos estão com dificuldades nestes conteúdos. Para compreensão linguística e matemática, é importante o desenvolvimento do raciocínio lógico.  Motivação e Justificativa: utilizando-se mecânicas dos jogos, pode-se motivar o usuário fique engajado a utilizar mais o programa e melhorar no mesmo, fazendo com que ele queira aprender seu conteúdo de forma eficiente.  Os capítulos de INTRODUÇÃO e 2. GAMIFICAÇÃO E MOTIVAÇÃO PARA APRENDER aprofundam a relevância do problema. |
| Artefato | Um aplicativo gamificado, que ensine história da computação e lógica de programação, utilizando-se de elementos de gamificação para motivação do usuário, utilizando-se de elementos presentes em trabalhos relacionados que serão apresentados mais adiante.  O artefato é proposto no Capitulo 4. PROPOSTA e desenvolvido no capítulo 6. DESENVOLVIMENTO, trabalhos relacionados no capítulo 3. ESTUDO EXPLORATÓRIO E TRABALHOS RELACIONADOS. |
| Busca da Solução | Para o desenvolvimento deste projeto, pesquisaram-se trabalhos que possuíam características que pudessem ser utilizadas para o ensino e motivação do usuário, não se encontrou nenhum trabalho que se trata sobre a história da computação, tornando-se um diferencial neste projeto.  Projetos Gamificados no capitulo 3. ESTUDO EXPLORATÓRIO E TRABALHOS RELACIONADOS. |
| Rigor da Pesquisa | Método de avaliação: segundo o DSR, será do tipo Observacional, utilizando-se de técnica Estudo de Campo, para monitorar o uso do artefato nos seguintes componentes curriculares, Lógica e Algoritmos com os cursos de Ciências da Computação e Sistemas de Informação e no Projeto Jovem Aprendiz.  Rigor da Pesquisa no capitulo7. PERCURSO DA PESQUISA. |
| Avaliação | A avaliação será realizada a partir da implantação do artefato nos seguintes componentes curriculares, Lógica e Algoritmos com os cursos de Ciências da Computação e Sistemas de Informação e no Projeto Jovem Aprendiz, utilizando-se de pesquisa, segundo a DSR, observacional com estudo de campo, monitorando o uso do artefato em nas disciplinas e projeto citados acima. Verificando a usabilidade e se o aluno considera um artefato que pode ser utilizado em sala de aula.  Avaliação no capitulo 7. PERCURSO DA PESQUISA e resultados no capítulo 8. RESULTADOS DA PESQUISA. |
| Contribuições da Pesquisa | Pode contribuir para projetos futuros na área de ensino-aprendizagem como referência de técnicas e bibliografia. |
| Comunicação da Pesquisa | A pesquisa será divulgada através da monografia aqui sendo desenvolvida e será disponibilizada para o público acadêmico através do site da biblioteca da instituição de ensino Feevale. |

# A Proposta

Este trabalho tem como objetivo desenvolver um aplicativo gamificado que auxilie no desenvolvimento do raciocínio lógico e computacional utilizando-se da temática da história da computação. Primeiramente foi realizado um estudo de sistemas que serviram de apoio e inspiração para a proposta.

## Estudo De Sistemas Para Apoiar A Proposta

Existem alguns aplicativos gamificados já presentes no mercado. Serão exemplificados alguns a partir de agora, o code.org[[9]](#footnote-9), Duolingo[[10]](#footnote-10) e o AppInventor[[11]](#footnote-11).

Os trabalhos apresentados abaixo e os trabalhos relacionados que serão apresentados mais adiante no texto, foram selecionados a partir da leitura de trabalhos referentes a gamificação, procurados no Google Acadêmico e sites de pesquisas cientificas como o Scielo. Selecionaram se projetos para a leitura e deles foram selecionados os projetos que foram testados e descritos abaixo.

O code.org é um site que possui diversos cursos de diversos níveis de dificuldade. Aqui nos concentraremos no curso chamado “Curso de 20 horas para Iniciantes”, mais especificamente no Curso Acelerado, 2ª ETAPA: O Labirinto. Nesta etapa deste curso, o usuário deverá criar uma sequência de comandos para que o pássaro percorra um labirinto e chegue ao porco. A tela é dividida em três partes essenciais, a primeira a esquerda é a representação do labirinto, onde o usuário pode visualizar se seus comandos estão sendo executados da maneira que ele os colocou para serem executados. A segunda parte que está presente no meio da tela e possui o título de Blocos, são os blocos de comando que o usuário pode realizar nesta etapa do labirinto. A terceira parte, que está à direita, é a sequência de comandos (blocos) que o pássaro irá seguir.

Para que o pássaro chegue ao porco, o usuário arrasta os comandos presentes na aba Blocos para a Área de Trabalho, formando uma sequência lógica de comandos para o pássaro seguir, quando a sequência está de acordo com o desejado, o usuário pode clicar no botão Executar presente na área do labirinto para verificar se a sua sequência de comandos faz com que o pássaro chegue ao porco. Caso isso não ocorra o usuário é informado para tentar novamente.

O labirinto possui um total de vinte “fases” que podem ser acessadas a qualquer momento, cada fase possui comandos novos e mais complexos e um labirinto novo para ser percorrido.

O Duolingo é um software que ensina línguas diferentes ao usuário através da seleção de línguas que o mesmo quer estudar, apresenta mecânicas como: traduza a frase, escute e escreva. Possui níveis de dificuldade de acordo com o nível de progresso do usuário. Sempre que propõe lições para o usuário, antes de apresentá-las, oferece dicas para facilitar o entendimento da lição.

Quando o usuário termina as lições do tópico, ele recebe uma nota de 0 a 5 de acordo com o domínio do tópico. O usuário possui desafios diários para realizar, como, ganhe 50 de experiência, sendo que cada lição para o usuário gera 10 de experiência. Depois de cumprir uma lição, o usuário recebe a porcentagem de domínio da língua na qual está estudando, podendo ver seu progresso geral, motivando-o a progredir ainda mais.

O AppInventor é uma aplicação desenvolvida pelo *Massachussets Institute of Technology* (MIT) nos Estados Unidos. Com ele é possível criar aplicativos de forma facilitada utilizando-se de blocos de comando, na tela de design você arrasta componentes de tela, como campos de texto e botões, ou componentes que não são vistos, como a possibilidade de chacoalhar o smartphone. Depois, na tela de blocos você define o que cada componente presente na tela de design fará, como, se o smartphone for chacoalhado a tela mudará de cor. Se o campo de texto for preenchido e o usuário clicar no botão que foi colocado o aplicativo falará aquela frase. Existem dezenas de comandos para o usuário criar um aplicativo da maneira que quiser.

O que estes softwares têm em comum? Todos têm o objetivo de ensinar e motivar o usuário. O code.org e AppInventor utilizam blocos de comando para facilitar o entendimento do usuário de como cada bloco funciona, uns englobando outros blocos dentro de si e outros fazendo referências através de flechas para outros blocos.

Já o Duolingo foca-se na motivação do usuário, transformando a plataforma de ensino em algo motivador através da gamificação, incitando o usuário a continuar utilizando a mesma, através de prêmios em medalhas, progresso em um assunto e proficiência já adquirida na plataforma através dos desafios propostos.

Os sistemas de apoio apresentados foram selecionados, pois possuem características de gamificação que foram utilizadas no projeto apresentado nesta monografia, como blocos de comando, resultados dos desafios e compensação em forma de estrelas para o usuário. A seguir será apresentada a proposta do aplicativo desenvolvido para aplicação em sala de aula.

## Proposta do aplicativo HistoCode

Decidiu-se realizar um trabalho que ensine história da computação e lógica de programação, pois não foi encontrado trabalho algum que possuísse esta motivação, e devido a muitos alunos desistirem da área de informática, principalmente, devido a não conseguirem desenvolver o raciocínio lógico necessário para prosseguirem na área. Este trabalho foca-se no ensino da história da computação e na lógica da mesma, não estando preso a uma linguagem específica, mas utilizando-se de linguagem estruturada para resolver os desafios.

Levando em consideração os objetivos do trabalho, foram levantados os requisitos do sistema. O aplicativo receberá o nome de HistoCode, será gamificado, ou seja, possuirá características que motivam o usuário, para isto o mesmo possuirá características da gamificação, como:

* Possuir regras;
* Objetivos a serem cumpridos;
* Ser interativo e de fácil uso;
* Demonstrar o resultado;
* Gratificar com estrelas/medalhas;
* Motivar a sempre melhorar;

O aplicativo possuirá regras e objetivos a serem cumpridos, pois cada uma das fases apresentadas possuirá seu próprio conjunto de comandos disponíveis, no qual uma sequência deles levará ao objetivo. Será de fácil uso, com fácil identificação do que pode ou não ser usado e como, facilitando a interação do usuário com o aplicativo. Os resultados serão mostrados ao usuário sempre que o mesmo selecionar para que a resolução proposta seja testada, indicando se está correta ou não. O ensino da lógica de programação possuirá os seguintes comandos:

* Pegar Entrada;
* Levar a Saída;
* Pule para;
* Copia para [posição];
* Copia de [posição];
* Some com [posição];
* Subtraia de [posição];
* Se zero;
* Se negativo;
* Aumente [posição];
* Diminua [posição];

Estes comandos foram selecionados, pois são comandos básicos da área de informática, muito dos mesmos presentes em todas as linguagens de programação, outros, como Aumente e Diminua são facilitadores, somente presentes para auxilio nesta aplicação. Além disso, são os comandos básicos necessários para o desenvolvimento de resoluções lógicas. Por ser simplificado, pode ser traduzido para qualquer linguagem de programação, se o usuário possuir algum conhecimento básico, assim facilitando para o professor/instrutor avançar o conteúdo. Estes blocos foram elaborados de maneira com que o usuário não seja impedido de progredir por falta de interpretação do comando, facilitando no desenvolvimento da lógica.

A cada fase será apresentada uma história na qual apresentará o problema a ser resolvido, o usuário deverá realizar uma listagem com as ações necessárias para que o problema seja resolvido. Quando o mesmo decidir que seu “programa” está pronto a ser rodado, o “representante” do usuário na fase realizará as ações escolhidas, ao final será mostrado o resultado positivo ou negativo do mesmo.

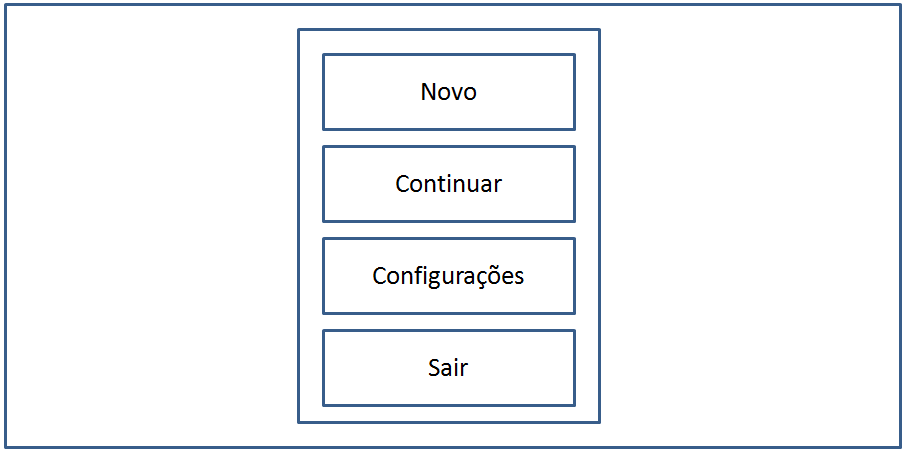
Decidiu-se criar um site para que possa ser possível criar desafios customizados, estes desafios podem ser criados tanto por professores/instrutores como por outros alunos, desde que o mesmo prove que o desafio é possível.

O desenvolvimento do aplicativo será realizado através da ferramenta Unity. Esta ferramenta é comumente utilizada para desenvolvimento de jogos para diversas plataformas, sendo este um dos motivos principais da utilização da mesma, pois assim é possível disponibilizar a utilização deste aplicativo através do site que será criado, não necessitando assim, a disponibilização através de APK (formato de um arquivo de instalação de aplicativos no sistema operacional Android), pois necessitaria de permissões especiais de instalação no dispositivo. Outro motivo da seleção da ferramenta foi por utilizar C# como linguagem de desenvolvimento e por possuir muitas das ferramentas necessárias para criar o aplicativo através de Arrastar e Soltar, facilitando assim a criação dos itens visuais.

## Design

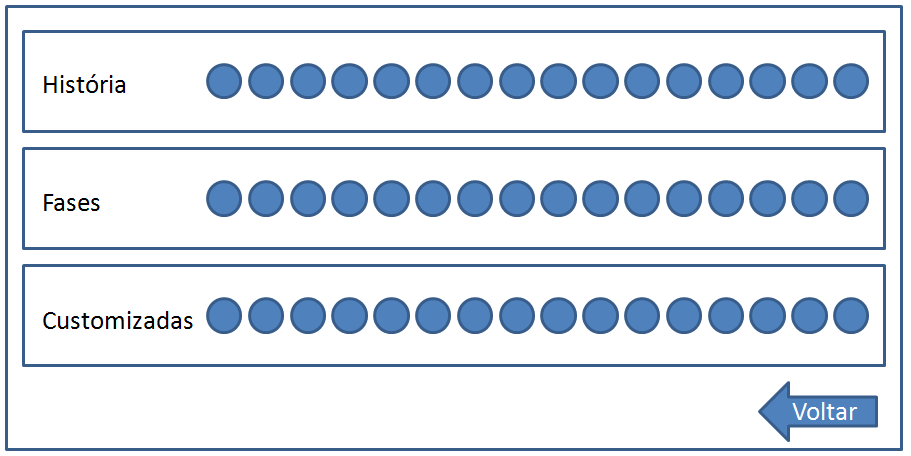
A seguir será apresentado como serão as telas mostradas para o usuário. A figura 4.1 representa a tela inicial do aplicativo, onde o usuário poderá criar um novo save, continuar algum já existente, acessar as configurações para, por exemplo, carregar o arquivo XML e sair do aplicativo.

Figura 4.1 – Tela Inicial



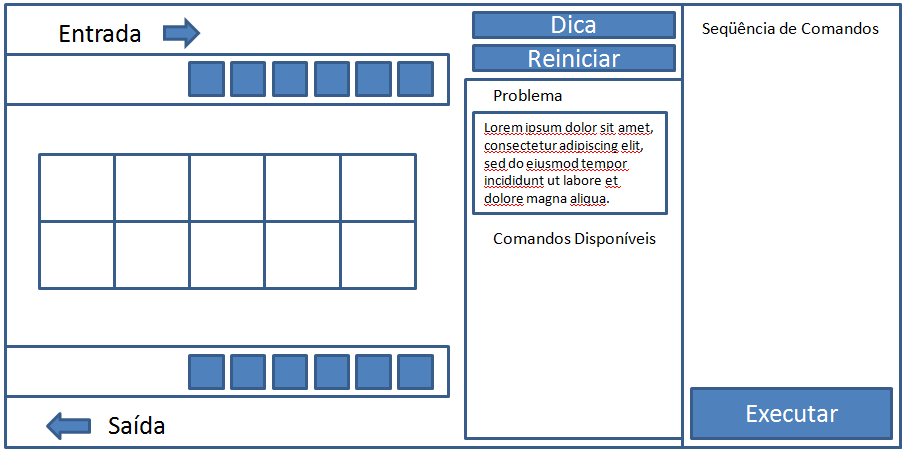
A figura 4.2 representa as fases que já existem no aplicativo, cada bolinha representando uma fase ou uma década da história da computação.

Figura 4.2 – Tela de Fases



A figura 4.3 representa a apresentação da fase para o usuário, possuindo a parte visual à esquerda, onde o usuário visualizará a sua sequência de comandos sendo executada. Na direita existirá a descrição do problema para o usuário, os comandos disponíveis e a área de sequência de comandos, onde o usuário criará sua solução.

Figura 4.3 – Tela Resolução de Fase



A figura 4.4 representa a demonstração dos resultados para o usuário, no caso apresentado, o usuário completou com êxito o problema, caso o usuário não tenha completado, ele poderá continuar de onde errou, não apresentando tela de resultados, mas sim uma representação na parte visual.

Figura 4.4 – Tela Resultados

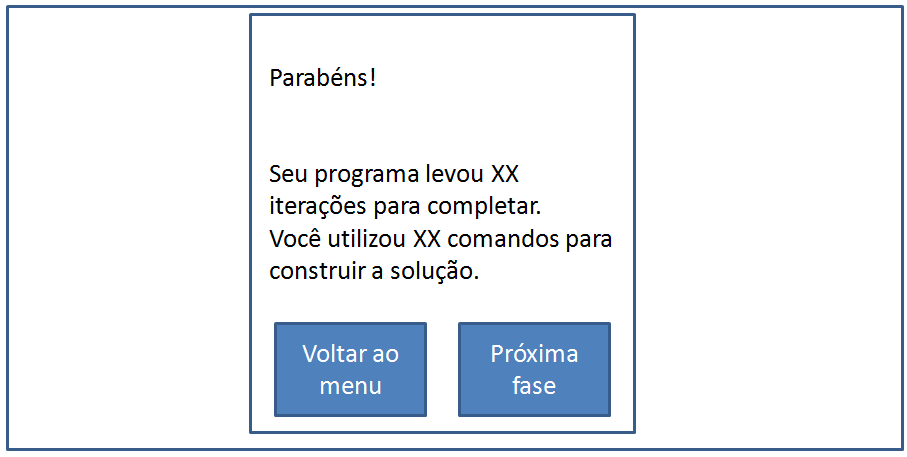
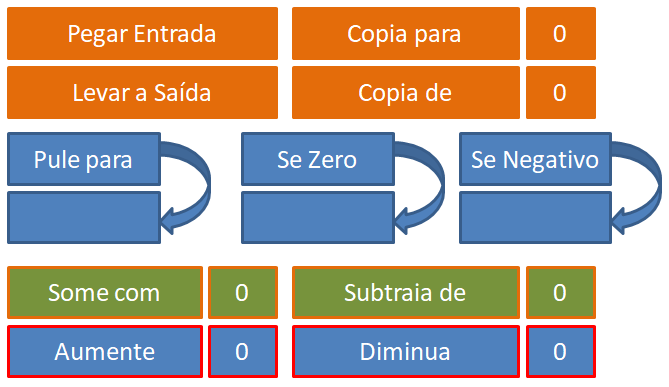


Figura 4.5 – Comandos



## História e Desafios

O aplicativo possuirá três áreas, a primeira área será concentrada na **história da computação**, a segunda serão as fases decididas pelo desenvolvedor do aplicativo, a terceira será dedicada aos desafios customizados criados pelos professores, instrutores e outros alunos, será a área em que o usuário poderá carregar um arquivo XML com novas fases e desafios para realizar, não necessitando ser sobre a história da computação, mas sobre o assunto que o desenvolvedor quiser.

Todas as fases poderão ser acessadas a qualquer momento, sem impedimentos por parte do aplicativo, abaixo será descrito como cada área funcionará.

A primeira área será composta, inicialmente por matérias jornalísticas, a seguir passando parar um computador com monitor CRT e em seguida para um monitor LCD, demonstrando que a tecnologia evoluiu e seus meios de comunicação também, estas matérias representarão como as novidades sobre a tecnologia daquele período eram divulgadas. Abaixo será descrita cada década. Se o usuário não quiser passar por esta etapa, os desafios presentes na segunda etapa possuíram referências aos acontecimentos presentes em cada época.

1. Década de 40: Bombe criado para decriptar o ENIGMA; Revelação do ENIAC; Primeiro programa a rodar em um computador.
2. Década de 50: Teste de Turing; Teclado utilizado como meio de entrada; Primeira imagem escaneada por computador; Jogo Tênis desenvolvido utilizando um osciloscópio.
3. Década de 60: IBM introduz a série 1400, substituindo sistema de tubos a vácuo por transistores; Braço Rancho, primeiro braço mecânico controlado por computador; Caixa Marrom, meio de jogar jogos de computador na televisão; Computador das missões Apollo é apresentado.
4. Década de 70: Kenbak-1 é vendido para escola privada; Intel produz o microprocessador, o Intel 4004; Surge o jogo PONG; Xerox PARC Alto, primeiro sistema a utilizar interface de janelas com o usuário, além de se comunicar com outros Altos; Apple-1 completado; Apple-II é comercializado, muito parecido com os computadores que possuímos hoje; Atari VCS, primeiro videogame, renomeado para Atari 2600.
5. Década de 80: IBM introduz o PC, computador que inflou o mercado de computadores pessoais; Commodore 64, reconhecido como computador mais vendido da história; Surge o CD-ROM; Apple lança o Macintosh, primeiro computador com mouse e interface gráfica a fazer sucesso; Nintendo libera o NES; Computador derrota mestre em Xadrez; Nintendo revela o Game Boy.
6. Década de 90: Primeiro SSD (Solid State Drive) surge; MQ-1 Predator Drone: primeiro veículo não tripulado a funcionar abertamente; Playstation é revelado pela Sony; Surge o DVD; Surge o Furby, primeiro robô de inteligência artificial a ser amplamente comercializado; Robô AIBO, robô de inteligência artificial que representava um companheiro canino.
7. Anos 2000: Primeiro celular com câmera; ASIMO, robô humanoide da Honda que podia andar, subir escadas e mudar de direção; Sony revela o Playstation 2; Surge o Pendrive; Microsoft libera o Xbox; Surge o Blu-ray; Opportunity e Spirit pousam em marte; World of Warcraft é lançado; Surge o Iphone; Surge o Minecraft.

A segunda área possuirá fases para conhecimento do usuário com relação aos comandos e possuirá um sistema de progressão básico utilizando os comandos apresentados anteriormente, a motivação ou o lúdico será que o usuário trabalha para uma empresa que manipula dados, estes chegam pela entrada e devem ser manipulados de acordo com o solicitado pela fase, levando o resultado a saída. Os primeiros desafios da segunda área serão somente de apresentação dos comandos disponíveis para o usuário, sendo elas:

1. Leve os itens da entrada para a saída. Comandos necessários: Pegar Entrada, Levar a Saída.
2. Utilize menos comandos para levar os itens da entrada para a saída. Comandos necessários: Pegar Entrada, Levar a Saída, Pule para.
3. Pegar duas entradas invertê-las utilizando “área de array” e levar resultados a saída. Comandos necessários: Pegar Entrada, Levar a Saída, Pule para, Copia para [posição], Copia de [posição].
4. Pegue duas entradas, some-as e leve o resultado a saída. Comandos necessários: Pegar Entrada, Levar a Saída, Pule para, Copia para [posição], Copia de [posição], Some com [posição].
5. Leve o que não for zero a saída. Comandos necessários: Pegar Entrada, Levar a Saída, Pule para, Se zero.
6. Para cada duas entradas, subtraia a primeira da segunda e leve o resultado a saída, após isso, subtraia o segundo do primeiro e leve a saída. Comandos necessários: Pegar Entrada, Levar a Saída, Pule para, Copia para [posição], Copia de [posição], Some com [posição], Subtraia de [posição], Se zero.

Os restantes dos desafios da segunda área terão como história e ou motivação referência para os itens vistos na primeira área, por exemplo, o Robô ASIMO necessita receber somente números positivos para que possa andar corretamente, leve a saída somente os números positivos. Nesta parte, todos os comandos estarão disponíveis para o usuário e as fases seguirão possuindo desafios que fazem referência a cada década apresentada anteriormente, os comandos são os apresentados no item 4.1, as fases serão as seguintes:

1. Quando o primeiro programa passou a rodar em um computador na década de 40, ele lia somente ZERO e UM de um cartão perfurado, por favor, leve a saída somente os números ZERO e UM.
2. O Teste de Turing foi criado para verificar se uma máquina é capaz de exibir comportamento inteligente equivalente a um ser humano. Por favor, leve a saída ZERO se a entrada for UM e UM se a entrada for ZERO.
3. Durante a década de 50, a primeira imagem foi escaneada por um computador, por favor, leve três vezes a saída o valor de entrada, representando o padrão RBG.
4. Durante a década de 60 a IBM introduziu a série 1400, computador empresarial para realização de cálculos, por favor, leve a saída o valor da entrada multiplicado por 3.
5. O braço Rancho foi o primeiro braço mecânico a ser controlado por computador, leve a saída somente números que não forem negativos, representando que o braço não acha algo para agarrar.
6. Na década de 70 surgiu o Xerox PARC Alto, primeiro sistema operacional a utilizar janelas com o usuário, além de se comunicar com outros Altos. Para representar os cálculos realizados para a comunicação, por favor, leve a saída o seguinte: O primeiro número será levado diretamente a saída, o segundo será diminuído do primeiro e lavado a saída, repita para o restante.
7. Na década de 70, o Apple-II começou a ser comercializado, para representar a fila de produção, leve a saída o valor da entrada multiplicado por si mesmo.
8. Na década de 80, a IBM introduziu o PC, que se tornou imensamente popular, representando a demanda crescente, por favor, leve à saída o valor da entrada multiplicado pelo número de itens da entrada já pegos, ou seja, primeiro item vezes 1, segundo item vezes 2 e assim por diante.

Estas fases foram organizadas desta maneira para possuírem uma progressão que o usuário possa acompanhar, fazendo com que o aprendizado seja mais fluído, sem prejudicar o usuário disponibilizando vários comandos de uma vez só para o mesmo.

A terceira área segue a mesma proposta de desafio da segunda área, porém as fases serão carregadas através de um arquivo XML, podendo o mesmo ser desenvolvido por outros usuários, sendo eles professores/orientadores ou alunos. Porém podem não possuir ligação com a história da computação, pois quem decidirá o tema é o criador do desafio. Este XML será gerado através de um site a ser desenvolvido, para que qualquer um possa desenvolver este arquivo sem conhecimento específico necessário.

Os tipos de gamificação que se utilizarão são a representação do desenvolvido, com o personagem realizando as tarefas de acordo com a sequência de comandos gerada. O sistema de feedback, ou seja, receber informações de acordo com o ocorrido, como quando o usuário erra uma mensagem aparece informando o erro. Outro sistema de feedback é a tela quando se concluiu um nível, nela está incluso o sistema de badges/estrelas, onde é possível visualizar o número de comandos utilizados, número de iterações realizadas pelo personagem e quantas estrelas foram obtidas naquele nível, assim motivando o usuário a melhorar e obter o número máximo de estrelas.

O arquivo será carregado através do menu principal da aplicação buscando o arquivo no próprio dispositivo. O XML seguirá o seguinte padrão:



Para cada tag<fase> será criada uma nova fase para o usuário resolver. <descricao> será o objetivo da fase que o usuário deverá resolver. As tags <entrada> e <saida> serão os valores de entrada que aparecerão para o usuário e a saída esperada pelo aplicativo para conclusão da fase. A tag <array> conterá as seguintes subtags <pos\_X>, que indicará quantas posições de array aparecerão para o usuário utilizar para desenvolver o seu programa. Nesta mesma tag poderá ser indicado se existe algum valor que deve aparecer desde o início para o usuário, exemplo: <pos\_4> 3 </pos\_4>, <pos\_5> 7 </pos\_5>.

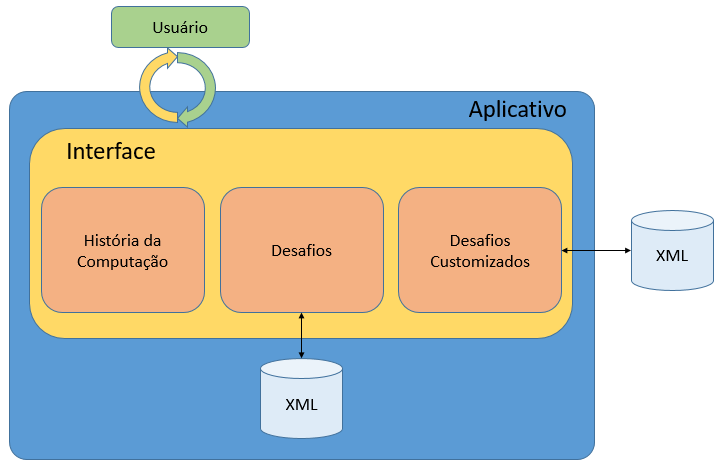
A tag <estatisticas> contém as estatísticas necessárias para que sejam dadas as estrelas para o usuário, sendo uma estrela sempre ganha ao completar a fase, uma por fazer o mesmo número ou menos de iterações e mesmo número ou menos de comandos.

Com a proposta apresentada, no capitulo a seguir será apresentado o modelo conceitual com relação as iterações do usuário com o aplicativo e como o aplicativo deve se comportar quando as informações forem solicitadas.

# Modelo conceitual

A seguir será apresentado o modelo conceitual deste projeto, o modelo contém somente a parte principal do aplicativo, que é a interação do usuário com a interface, a mesma contém a visualização da História da Computação e a visualização dos Desafios Normais e Customizados. Os Desafios são buscados em um arquivo XML dentro do aplicativo, não podendo ser modificado externamente, os Desafios Customizados são buscados em um arquivo separado fora do aplicativo, podendo ser alterado externamente ou através do carregamento de um arquivo com novos desafios na própria interface do aplicativo.

Figura 5.1 – Modelo Conceitual Simples



## UML

### Diagrama de Casos de Uso

Figura 5.2 – Caso de Uso

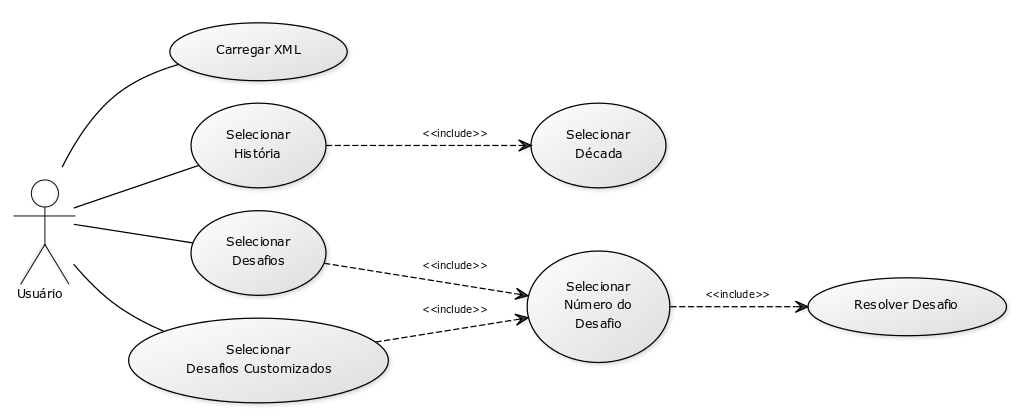


Tabela 5.1 – Selecionar História

|  |  |
| --- | --- |
| Nome do caso de uso | Selecionar História |
| Descrição Breve | Selecionar a opção História da Computação. |
| Fluxo | 1. Abrir aplicativo; 2. Selecionar História da Computação; |
| Condições Prévias | Nenhuma |

Tabela 5.2 – Selecionar Década

|  |  |
| --- | --- |
| Nome do caso de uso | Selecionar Década |
| Descrição Breve | Selecionar a que gostaria de aprender sobre. |
| Fluxo | 1. Selecionar uma das opções de década; |
| Condições Prévias | Selecionar Slot |

Tabela 5.3 – Selecionar Desafios

|  |  |
| --- | --- |
| Nome do caso de uso | Selecionar Desafios |
| Descrição Breve | Selecionar a opção desafios. |
| Fluxo | 1. Abrir aplicativo; 2. Selecionar Desafios; |
| Condições Prévias | Nenhuma |

Tabela 5.4 – Selecionar Desafios Customizados

|  |  |
| --- | --- |
| Nome do caso de uso | Selecionar Desafios Customizados |
| Descrição Breve | Selecionar a opção desafios customizados. |
| Fluxo | 1. Abrir aplicativo; 2. Selecionar Desafios Customizados; |
| Condições Prévias | Nenhuma |

Tabela 5.5 – Selecionar Número do Desafio

|  |  |
| --- | --- |
| Nome do caso de uso | Selecionar Número do Desafio |
| Descrição Breve | Selecionar desafio a ser resolvido. |
| Fluxo | 1. Selecionar o número do desafio; |
| Condições Prévias | Selecionar Slot |

Tabela 5.6 – Resolver Desafio

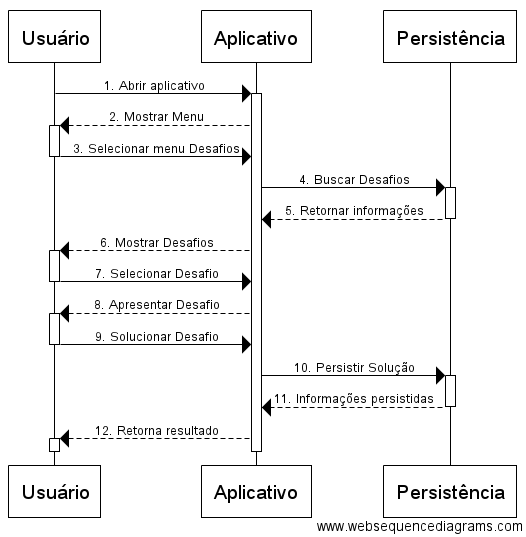
|  |  |
| --- | --- |
| Nome do caso de uso | Resolver Desafio |
| Descrição Breve | Resolução da história proposta da fase através de comandos de linguagem estruturada; |
| Fluxo | 1. Prestar atenção à história proposta; 2. Resolver a história através de comandos de linguagem estruturada predefinidos até a fase selecionada; |
| Condições Prévias | Selecionar fase |

Tabela 5.7 – Carregar XML

|  |  |
| --- | --- |
| Nome do caso de uso | Carregar XML |
| Descrição Breve | Carregar um XML de fases que irá adicionar fases novas para o usuário em uma nova seção na seleção de fase; |
| Fluxo | 1. Selecionar opções no menu principal; 2. Selecionar arquivo XML presente no dispositivo; 3. Aguardar confirmação do carregamento; |
| Condições Prévias | Nenhuma |

### Diagrama de Sequência

Figura 5.3 – Diagrama de Sequência



O diagrama de sequência acima representa a interação do usuário para abrir o aplicativo até o ponto em que ele conclui uma fase do mesmo, não incluindo a sequência de carregamento de XML e nem a de visualização da história por serem muito simples. A seguir será descrito o que acontece em cada etapa representada acima.

Tabela 5.5 – Descrição diagrama de sequência

|  |  |
| --- | --- |
| **Etapa** | **Descrição** |
| 1 | Abrir a aplicação. |
| 2 | Aplicativo retorna o menu de opções. |
| 3 | O usuário seleciona o menu Desafios. |
| 4 | O aplicativo comunica-se com a persistência buscando as informações dos desafios disponíveis. |
| 5 | Persistência retorna para o aplicativo as informações dos desafios. |
| 6 | Aplicativo mostra os desafios disponíveis para o usuário. |
| 7 | Usuário seleciona o desafio que deseja realizar. |
| 8 | O aplicativo retorna a tela com os desafios e os blocos de comando para solução. |
| 9 | Usuário cria sua sequência de blocos e diz para o aplicativo testar. |
| 10 | Aplicativo salva as informações sobre a sequência de comandos. |
| 11 | Persistência informa para o aplicativo que as informações foram salvas. |
| 12 | Após o processamento da sequência de comandos, é passado para o usuário se a sua solução foi satisfatória para a resolução do desafio. |

Com a proposta apresentada, no capitulo a seguir, será colocado como o desenvolvimento ocorreu, as modificações necessárias que foram realizadas com relação à proposta e o artefato final do mesmo.

# Desenvolvimento

Como dito anteriormente, o desenvolvimento do aplicativo ocorreu em Unity, por utilizar C# como linguagem de programação, possuir ferramentas que auxiliam na criação da parte visual e possuir bibliotecas que facilitam o desenvolvimento de movimentação do personagem e dos blocos de comando.

Na proposta pensava-se em utilizar o sistema MVC (Model View Controller), porém durante os estudos e amadurecimento do conhecimento da plataforma, descobriu-se que o mesmo já possuía integrado esse sistema. A View no Unity são as Scenes, onde o conteúdo é apresentado para o usuário, ou seja, a parte visual da aplicação. A Controller é as configurações de Build (Build Settings) presente no Unity, nela são decididas quais Scenes poderão ser apresentadas para o usuário, se uma Scene não estiver nesta configuração, à mesma não poderá ser chamada. Já a Model são as classes que associamos a uma Scene.

Ao planejar a arte do aplicativo, viu-se que o que havia sido proposto era muito sobrecarregado para o usuário, então se decidiu modificar as telas para serem mais “User friendly”, ou seja, mais amigáveis ao usuário.

O menu principal foi modificado, como o aplicativo foi compilado para ser utilizado pela internet através do site criado, foi deixado somente a partir da seleção da história ou desafios, removeu-se as linhas com bolas para a seleção de fases e separou-se a seleção da História da Computação dos Desafio, cada um tendo uma sua própria tela. Cada fase recebeu um botão, nele e ao redor dele são apresentadas informações como o número da fase e o número de estrelas já obtidas na mesma.

A tela seleção de fases foi alterada para maior facilidade de uso do usuário, anteriormente a mesma possuía muitas informações aglomeradas e de difícil utilização para o usuário, foi alterado para que só existam 8 botões de seleção de fase visíveis na tela, facilitando o uso e inserindo o sistema de estrelas, para que o usuário saiba se pode ou não melhorar naquela fase. As fases continuarão sendo lidas de um XML, porém as fases que já vierem com o aplicativo não poderão ser modificadas externamente, ou seja, o arquivo XML destas fases será compilado juntamente do aplicativo. Quando o aplicativo for aberto, as fases serão carregadas em memória, e quando uma fase for selecionada, a mesma será carregada da memória, diminuindo o consumo de processamento do smartphone. As fases seguiram o que que foi definido na proposta, sendo as 6 primeiras fases de tutorial, para aprender a utilizar o aplicativo, as fases seguintes utilizam a história da computação como motivação do desafio.

As fases customizadas foram retiradas da versão final de desenvolvimento, pois é uma funcionalidade mais focada na interação professor/aluno, não sendo necessária para o teste com o usuário. A seguir imagem da versão final da seleção de fases.

Figura 6.1 – Tela seleção de fases



Quando uma fase do tipo história for carregada, anteriormente seria uma animação pausada, na qual apareceria uma das novidades tecnológicas daquela década, apresentando o acontecimento e um texto sobre o mesmo, quando o usuário tocasse na tela apareceria à próxima novidade. Durante o desenvolvimento decidiu-se caracterizar cada década, passando a ser o meio de comunicação principal da época, começando pelo jornal, passando para o um monitor CRT e finalmente para um monitor LED/LCD. A seguir as telas da versão final de cada uma das apresentações dos eventos históricos.

Figura 6.2 – História no Jornal



Figura 6.3 – História no monitor CRT



Figura 6.4 – História no monitor LED/LCD



Na tela de solução de desafios, decidiu-se modificar a disposição dos menus, pois anteriormente havia muita informação na tela para o usuário. A tela seguiu o padrão definido na proposta, a proposta segue o modelo utilizado pelos projetos apresentados nos trabalhos relacionados, possuindo três áreas principais, uma parte de representação da solução, uma com os comandos disponíveis e outra com a solução criado pelo usuário. A história e os erros da fase foram retirados da tela e serão apresentadas em uma alto-falante que o usuário pode tocar. A fase começará com este alto-falante funcionando e apresentando o problema, o usuário tocará nele para ver a apresentação novamente. Os comandos disponíveis no nível ficam sempre disponíveis a esquerda da área onde a sequência de comandos é criada (área verde escura a direita). Também seguindo o modelo apresentados pelos trabalhos relacionados, utilizou-se uma flecha para representar qual comando está sendo executado no momento para o usuário. A seguir imagens mostrando a versão final da tela de fases e seus menus.

Figura 6.5 – Tela de resolução de desafio



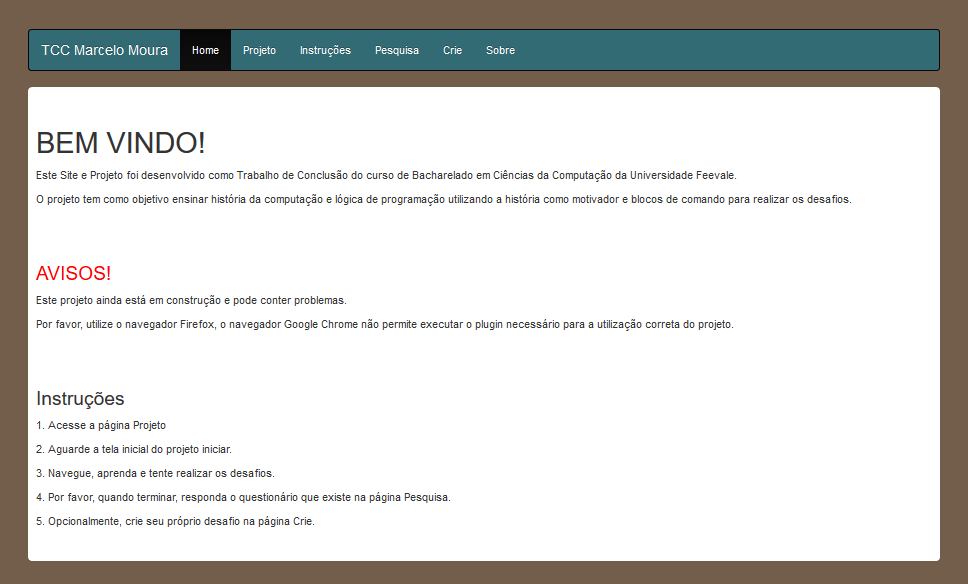
Quando o usuário comete algum erro de execução, uma informação é lançada pelo autofalante, informando que o comando não pode ser realizado com as mãos vazias, que o valor esperado era outro do que foi levado a saída entre outros possíveis erros. Quando o usuário completa o nível, aparece para o mesmo suas estatísticas, como número de comandos utilizados e número de iterações realizadas pelo personagem do jogo. A tela de resultado possui três estrelas, uma sendo ganha por completar o nível, outra por utilizar o mesmo ou menos números de comandos estabelecidos e outra pelo mesmo ou menor número de iterações do usuário. As estrelas aparecem na tela de seleção de níveis, deixando visível para o usuário se ele pode ou não melhorar naquele nível. A seguir a tela de conclusão do nível.

Figura 6.6 – Tela de resultados



O site tem como objetivo neste iteração, servir como distribuição do projeto, tirar dúvidas sobre a utilização dos comandos e meio de responder ao questionário, também servirá como ferramenta para a criação dos desafios customizados. O site foi desenvolvido em HTML[[12]](#footnote-12), que é uma linguagem de programação usada para criação de páginas Web, utilizando CSS[[13]](#footnote-13) para adicionar estilo ao documento e Javascript[[14]](#footnote-14), que é uma linguagem de programação que faz com que não seja necessário que a informação passe pelo servidor, para a verificação da fase que pode ser criada pelo usuário. O site possui uma tela inicial, nela contém informações sobre o porquê do desenvolvimento do mesmo, avisos sobre possíveis problemas e as instruções para a utilização dos usuários. Para acessar o site criado e o projeto, acesse www.tccmarcelomoura.com. A seguir a tela inicial do site.

Figura 6.7 – Site



A tela Home contém informações de como executar o projeto e instruções básicas sobre como o usuário deve navegar pelo site.

A tela Projeto contém o aplicativo para que o usuário possa utilizar o mesmo, o aplicativo foi compilado em WebGL[[15]](#footnote-15), que é uma biblioteca Javascript que permite a visualização de itens 2D e 3D no navegador, tecnologia pertencente ao HTML5, através da Unity[[16]](#footnote-16), programa para criação de jogos 2D e 3D, facilitando a distribuição e utilização dos usuários, facilitando a realização da pesquisa que será apresentada a seguir.

A tela Instruções contém informações sobre a utilização de cada comando e instruções gerais de como o aplicativo funciona, serve como referência para dúvidas surgidas sobre os comandos, explicando seu funcionamento e como deve e pode ser utilizado.

A tela Pesquisa contém um formulário de pesquisa sobre a utilização da aplicação, utilizou-se a ferramenta Typeform[[17]](#footnote-17) para isto, pois a mesma possui visual atraente e os resultados da mesma são facilmente filtrados através da ferramenta.

A tela Crie é onde o usuário, professor ou instrutor pode criar desafios customizados para serem distribuídos para outros usuários ou alunos, a tela utiliza Javascript para realizar a verificação se o desafio é possível e gerar o arquivo XML para download. A tela possui todos os campos necessários, porém não possui o visual do aplicativo, sendo somente um formulário web a se preencher.

No capitulo a seguir será colocado como a pesquisa se realizará, com quem será feita, quais seus questionamentos e seus resultados.

# Percurso da Pesquisa

Este projeto seria aplicado através do projeto Logicando, projeto desenvolvido por estudantes e professores da Universidade Feevale em colaboração com as escolas locais de Novo Hamburgo. O trabalho tem como objetivo desenvolver o raciocínio lógico dos alunos nos anos finais do ensino fundamental. Pretendendo-se utilizar o aplicativo proposto aqui nas oficinas 1A, porém devido a datas não adequadas o aplicativo foi testado em sala de aula nos seguintes componentes curriculares, Lógica e Algoritmos com os cursos de Ciências da Computação e Sistemas de Informação, pois ainda se enquadra no objetivo do projeto.

Segundo a DSR, o projeto foi desenvolvido como pesquisa aplicada, sendo exploratória e descritiva, utilizando-se de pesquisa de estudo de campo. A pesquisa é aplicada, pois ela deve gerar conhecimento e através dele resolver problemas específicos, no caso deste projeto, foi gerado o conhecimento de como aplicar um aplicativo gamificado para melhorar o ensino de lógica de programação e história da computação.

É exploratória, pois ela se encontra em fase inicial, tendo como objetivo proporcionar conhecimento sobre o assunto, gerando sua definição e delimitações. Descritiva, pois explica e interpreta o que ocorre durante o desenvolvimento do projeto sem interferir diretamente no que está acontecendo. Estudo de campo utilizando-se de pesquisa com o usuário, após a utilização do aplicativo e pesquisa observacional verificando a utilização do aplicativo visto pelo aplicador.

Após o uso do aplicativo, foi solicitado que o usuário respondesse um questionário sobre a utilização do aplicativo. A pesquisa foi respondida pelos alunos que utilizarem o aplicativo. O questionário seguiu o modelo Lickert de pesquisa, onde as perguntas são respondidas através de níveis de satisfação, sendo o pior caso Discordo Plenamente e o melhor Concordo Plenamente.

A pesquisa do aluno foi focada se o aplicativo foi capaz de lhe ensinar lógica de programação e história da computação, se o aplicativo foi de uso satisfatório e se o mesmo acredita que um software pode ser utilizado como ferramenta de ensino em vários níveis do ensino e em vários conteúdos. Ver pesquisa no Apêndice A.

Para a pesquisa descritiva, foi utilizado um questionário que será respondido pelo aplicador do aplicativo em sala de aula. É focado na observação da utilização do aplicativo pelo usuário. Verificando diversos pontos como, se o usuário utilizou facilmente os menus, ou conseguiu adaptar-se a interface do mesmo, se houveram muitas dúvidas e pontos de melhoria. O questionário possui na parte esquerda a pergunta a ser respondida e uma área para observações naquela pergunta. A direita o nível que o aplicador considera que aquela pergunta está durante a aplicação em sala de aula. Os níveis são: Insuficiente, Suficiente, Regular, Bom e Muito Bom. Ver questionário observacional no Apêndice.

Com os resultados das pesquisas foi verificado se é de interesse público utilizar aplicativos e softwares como ferramenta de ensino para manter o interesse do aluno no ensino e fornecer as informações necessárias para que o aluno aprenda o conteúdo e não somente o decore e se o aplicativo criado serviria como ferramenta de auxílio ao ensino.

# Resultados da pesquisa

A pesquisa foi realizada com os componentes curriculares de Lógica e Algoritmos, onde as turmas eram compostas por alunos dos cursos de Ciências da Computação e Sistemas de Informação, e no Projeto Jovem Aprendiz. No total foram 27 questionários respondidos pelos alunos de Lógica e Algoritmos e 55 respondidos pelo Jovem Aprendiz, estes serão analisados abaixo.

Inicialmente serão analisados os resultados sobre a História da Computação. Primeiramente foi perguntado se o usuário possuía conhecimento sobre a história da computação, nas disciplinas, 85% possuía algum conhecimento e 7% não possuía nenhum e no Jovem Aprendiz, 80% possuía algum e 11% não possuía. Depois foi perguntado se após o uso do aplicativo, o usuário havia aprendido coisas novas sobre a história, nas disciplinas, 63% concordaram que haviam aprendido novas coisas, 33% foram indiferentes e 4% discordaram, no Jovem Aprendiz, 84% aprenderam coisas novas 13% foi indiferente e 3% discordaram.

Figura 8.1 – Gráfico conhecimento sobre a história da computação disciplinas.

Figura 8.2 – Gráfico conhecimento sobre a história da computação Jovem Aprendiz.

Figura 8.3 – Gráfico aprendi sobre a história da computação disciplinas.

Figura 8.4 – Gráfico aprendi sobre a história da computação Jovem Aprendiz.

Como pode ser visto, a história da computação atingiu 63% dos usuários nas disciplinas sendo que desses, somente um não possuía conhecimento sobre e o restante possuía algum. No Jovem Aprendiz, 80% foi atingido, sendo que somente 2 usuários discordaram sobre ter aprendido algo novo, estes disseram possuir bastante conhecimento sobre a história. Com isso pode-se dizer que o aplicativo pode ser utilizado como meio de ensinar a história da computação, podendo gerar conhecimento para cerca de dois terços dos usuários.

Agora será analisado os desafios de lógica. Primeiramente foi perguntado se o usuário considerava que resolvia problemas de lógica facilmente, nas disciplinas, 74% respondeu que sim. Porém, após a utilização do aplicativo, quando os usuários foram perguntados se se consideravam mais aptos a resolver problemas de lógica, somente 38% concordaram, 48% foram indiferentes e o restante discordou. No Jovem Aprendiz, 76% respondeu que não resolvia problemas de lógica facilmente, quando perguntados se considerava estar mais apto, 47% concordou, 22% foi indiferente e 31% discordou.

Verificando os dados sobre a utilização do aplicativo e a clareza dos desafios, muitos destes que foram indiferentes ou discordaram também disseram que a utilização estava ruim e os objetivos não estavam claros, no caso do Jovem Aprendiz, observou-se que muitos que tiveram dificuldade na resolução, desistiram facilmente ou nem tentaram resolver os problemas. Resultados das pesquisas abaixo.

Figura 8.5 – Gráfico resolvo problemas de lógica facilmente disciplinas.

Figura 8.6 – Gráfico resolvo problemas de lógica facilmente Jovem Aprendiz.

Figura 8.7 – Gráfico considero que estou mais apto disciplinas.

Figura 8.8 – Gráfico considero que estou mais apto Jovem Aprendiz.

Verificando a usabilidade do aplicativo, nas disciplinas, ficou claro que 41% teve clareza na utilização, 30% teve dificuldades no uso e 29% foi indiferente, no Jovem Aprendiz, somente 37% teve clareza, 33% foi indiferente e 31% discordou. Durante o processo observacional, verificou-se que os problemas encontrados na utilização do aplicativo aconteciam na realização dos desafios, devido à pouca informação de como os comandos funcionavam e no caso do Jovem Aprendiz, novamente, os usuários que não realizaram os problemas por não conseguirem utilizar ou desistirem do aplicativo. Mais sobre o processo observacional adiante. Confirmando que houveram problemas na realização dos desafios, podemos verificar no gráfico 8.11 e 8.12 sobre informações claras para realização dos desafios que, nas disciplinas, 37% considerou que as informações eram insuficientes, 33% foram indiferentes e somente 30% consideraram informações suficientes. No Jovem Aprendiz, 45% consideraram as informações claras, 33% indiferente e o restante discordou. Esta diferença de informações claras deveu-se ao fato de modificações realizadas entre a aplicação nas disciplinas e no Jovem Aprendiz, além disso, no segundo, decidiu-se realizar uma explicação sobre o programa e sua utilização, enquanto nas disciplinas o usuário que devia utilizar e entender, sem qualquer explicação inicial de uso.

Figura 8.9 – Gráfico fácil utilização do aplicativo disciplinas.

Figura 8.10 – Gráfico fácil utilização do aplicativo Jovem Aprendiz.

Figura 8.11 – Gráfico informações claras disciplinas.

Figura 8.12 – Gráfico informações claras Jovem Aprendiz.

Mesmo com estes problemas para a resolução dos desafios, nas disciplinas, 82% concordam que a ferramenta apresentada cumpriu o dever como uma ferramenta de auxilio para o aprendizado e 92% concordam que esta ferramenta, ou como esta, pode ser utilizada como auxilio no ensino. No Jovem Aprendiz, 64% concordam que a ferramenta apresentada cumpriu seu dever e 29% foi indiferente. 78% concordam que esta ferramenta, ou como esta pode ser utilizada como auxilio no ensino.

Os altos números dos indiferentes do Jovem Aprendiz são devido ao fato citado de que alguns usuários desistiram da utilização por não conseguirem realizar os objetivos. Mesmo assim, esta pesquisa abre novas oportunidades para o desenvolvimento de ferramentas que possam ser utilizadas em sala de aula, devido ao grande número de usuário atingidos e que concordam que uma ferramenta como esta pode ser utilizada em sala de aula.

Figura 8.13 – Gráfico auxilio para o meu ensino disciplinas.

Figura 8.14 – Gráfico auxilio para o meu ensino Jovem Aprendiz.

Figura 8.15 – Gráfico ferramenta como auxilio ao ensino disciplinas.

Figura 8.16 – Gráfico ferramenta como auxilio ao ensino Jovem Aprendiz.

Passando para a análise observacional, foi verificado o uso do site pelo usuário e o uso do aplicativo. Com relação ao site, não houve nenhum problema na utilização do mesmo, tanto na navegação quanto na parte da pesquisa. Na parte do aplicativo, verificou-se que os menus estavam sendo percorridos sem problemas, a visualização da história da computação foi natural e prendeu a atenção do usuário. A resolução dos desafios foi onde os usuários tiveram dificuldades.

Os usuários tiveram muita dificuldade no início para entender como se criava uma sequência de comando e como os mesmos funcionavam. Alguns usuários tiveram dificuldade na interpretação dos desafios. Utilizou-se o momento para verificar bugs de software para correção futura.

No final do questionário com o usuário havia um campo para críticas, sugestões e observações sobre o projeto, este campo não era obrigatório. Grande parte das respostas nestes campos eram sobre melhorar a explicação dos desafios e outros colocaram que os comandos deveriam ser melhor explicados, pois não estava claro o seu funcionamento. Um usuário colocou que mesmo com estes problemas citados acima, uma ferramenta como esta iria auxiliar muitos em disciplinas como a de Lógica, onde o mesmo possui dificuldades, assim aprendendo de forma mais rápida ou para aplicação de avaliações em sala de aula. Outras sugestões que surgiram foram de um botão para limpeza dos comandos já colocados na área de solução e um botão para ir diretamente para o próximo desafio.

Com os dados apresentados, podemos verificar que apesar dos problemas, muitos dos usuários consideraram que aprenderam mais sobre a história da computação e que esta ferramenta, ou como esta, poderia ser utilizada como ferramenta de auxilio no ensino. Porém os dados gerais não foram satisfatórios, pois os desafios, que são a área de maior interesse para o desenvolvimento do aluno, possuiu pouca aprovação, atingindo satisfatoriamente cerca de um terço dos usuários somente.

Pela baixa aprovação dos desafios, viu-se que será necessário alterar o que foi passado de sugestões dos usuários, como melhorar a explicação do desafio, melhorar o entendimento dos comandos e melhorar a navegação entre as fases.

Com relação aos objetivos do trabalho, o objetivo geral do mesmo, que é “Desenvolver um aplicativo gamificado que auxilie no desenvolvimento do raciocínio lógico e computacional usando como temática a história da computação”, foi realizado, porém a temática não estava bem integrada aos desafios e os mesmos possuíam dificuldade de entendimento.

Nos objetivos específicos, o objetivo “Identificar os elementos de raciocínio lógico e da história da computação necessários para que o usuário tenha o mínimo de lógica necessária para progredir satisfatoriamente na área da informática” foi realizado, foram identificados os elementos citados, porém foram mal executados, ainda sendo necessário melhorar este ponto.

“Propor sistema que elenque raciocínio lógico e história da computação” foi cumprido, porém, como já dito, foi passado para o usuário de maneira insatisfatória, necessitando de mais trabalho.

“Avaliar se o sistema pode ser utilizado como ferramenta de ensino de lógica e história da computação através de pesquisa com o usuário”, foi avaliado, os usuário consideraram que esta ferramenta, ou como esta pode ser utilizada como auxilio em sala de aula, porém ainda necessita de modificações para a plena utilização da mesma em sala de aula. Mais sobre os trabalhos futuros adiante.

# CONCLUSÃO

Pode-se constatar, pelas pesquisas realizadas, que desenvolver um aplicativo gamificado voltado para a educação deve ser muito bem estudado, para que seja intuitivo e de fácil uso do usuário. O aplicativo deve repassar o conteúdo necessário e possuir as características presentes em jogos, como: ser divertido, motivar, possuir desafios entre outros para que o usuário se prenda ao aplicativo, querendo aprender mais e melhorar seu conhecimento, buscando mais informações dentro e fora de sala de aula.

Durante o estudo teórico, foi visto que certas características da gamificação devem estar presentes para que o usuário fique motivado para progredir, como: uma história como tema, feedback para o usuário com suas informações e as regras a serem seguidas. A história da computação foi utilizada neste trabalho como o motivador, sendo apresentada incialmente sem o contexto dos desafios e após os níveis introdutórios como um meio motivador. O usuário recebia constantemente feedback de seus erros e quando completava o nível recebia as informações de seu desempenho, sendo informado se poderia melhorar ou não. A cada nível o usuário possuía um desafio diferente, ou seja, regras diferentes, motivando o mesmo a continuar.

No geral, o aplicativo se mostrou de fácil uso na navegação, atingiu dois terços dos usuários na parte de história da computação, ensinando elementos novos sobre a história da computação, porém os desafios tiveram somente um terço de aprovação, tendo problemas para passar o conteúdo inicialmente, mas quando o usuário entendia o que devia ser feito e/ou como os comandos funcionavam o desafio era realizado com facilidade. Sendo este a grande parte das críticas sobre o aplicativo, a dificuldade encontrada no entendimento.

O site não teve problemas para passar o conteúdo para o usuário, sendo de fácil navegação e provendo informações sobre os comandos em caso de dúvida. Como o aplicativo foi colocado no site, ficou muito mais fácil o acesso para o usuário, não sendo necessário a utilização de um tablet ou smartphone para cada um. Como a pesquisa estava presente no site também, não sendo um questionário físico, mas sim interativo, muitos dos alunos, ao final do período da aplicação, respondiam rapidamente ao mesmo.

Os trabalhos relacionados que serão apresentado adiante tiveram grande impacto no design do aplicativo, seguindo o padrão de tela dos mesmos e utilizando blocos de comando para a realização das tarefas. Os blocos foram alterados de maneira que ficasse mais próximo de uma linguagem estruturada do que de uma linguagem de programação, assim estimulando a lógica e não ensinando uma linguagem em si.

O diferencial do HistoCode foi a maneira como foi abordado, não seguindo uma linguagem de programação especifica e nem sendo muito caricato, ou seja, não utilizava de analogias como plante, corte, utilizando-se de blocos para representar o que o personagem deveria fazer, fazendo comparação com os trabalhos relacionados apresentados. Utilizou-se da história da computação como motivador, pois nenhum outro trabalho encontrado a utilizava e por ser exatamente o foco dos alunos que se decidiu utilizar para a pesquisa, ingressantes das áreas da computação que muitas vezes não conhecem a história de como possuímos os eletrônicos hoje presentes na vida de todos. Este trabalho pode servir como referência tanto bibliográfica quanto para desenvolvimentos futuros na área, sobre erro e acertos cometidos durante o desenvolvimento e aplicação. A seguir os trabalhos relacionados a este projeto.

## Contribuição do trabalho frente a trabalhos relacionados

Nesta seção serão apresentados trabalhos que possuem uma utilização parecida com que a foi apresentado nos capítulos PROPOSTA e DESENVOLVIMENTO. Ao final será realizado uma comparação destes trabalhos com o proposto nesta monografia.

No site code.org[[18]](#footnote-18), na área Hora do Código, existem diversas atividades na qual o usuário pode selecionar uma temática e criar soluções para os desafios, todas as atividades possuem o mesmo padrão de tela, já citado anteriormente, possuindo uma área de código e uma área de representação da atividade, onde é demonstrado o resultado do que foi criado em código. Serão apresentados dois casos a seguir de atividades da Hora do Código.

O primeiro caso é o Minecraft Hour of Code, o mesmo possui a temático do jogo Minecraft, da empresa Mojang. O objetivo desta atividade é completar desafios relacionados ao jogo, como fugir de inimigos, construir sua casa e plantar sua comida, tudo isto utilizando blocos de comando, estes bloco possuem ações, como: Avance, Repita, Plante e muitas outras, com isto, o usuário cria a sua solução para o problema e o executa, verificando se o personagem seguiu os comandos de forma necessária a concluir o desafio. Quando a solução é executada o comando sendo executado no momento é destacado, mostrando para o usuário qual está sendo executado no momento. A seguir uma imagem de como é a representação do jogo e sua interface.

Figura 8.1 – Interface Minecraft Hour of Code



O segundo caso é o do jogo CodeCombat, nele o usuário é um personagem que deve percorrer uma masmorra cheia de inimigos para coletar tesouros. Utiliza de comandos em Javascript ou outra linguagem selecionada para criar a sequência necessária que o personagem deve seguir, possui regras, como Evite Espinhos, Colete todas as Joias e comandos predefinidos a serem utilizados em cada fase. Quando está sendo executado, a linha de comando que está sendo executada é destacada com uma flecha, indicando o que é aquele comando que está sendo executado. A seguir uma imagem da representação visual do jogo e sua interface.

Figura 8.2 – Interface CodeCombat



A partir dos sistemas apresentados, este trabalho buscou inspiração para a proposta de aplicativo, como o sistema de blocos de comandos e a gamificação, logo proporá para o usuário como desenvolver, demonstrando como cada bloco de comando funciona, deixando livre para que o usuário crie sua própria solução, ao final apresentará os resultados do desenvolvido, como número de iterações e número de comandos.

Com estes projetos apresentados, porque não utiliza-los ao invés da proposta que será apresentada a seguir? Estes projetos possuem temáticas populares e bem difundidas, a proposta que será apresentada tem como temática a história da computação, algo que foi pouco explorado, e tem como objetivo ensinar a lógica de programação, utilizando-se de comandos simples, desenvolvendo assim um raciocínio lógico que pode ser utilizado em qualquer meio, sendo ele da computação ou não.

Com o HistoCode e trabalhos relacionados apresentados, podemos definir que um jogo é focado em um objetivo específico que não pode ser representado ou passado ao usuário fora do mesmo, como, por exemplo, sobreviva ao apocalipse. Um aplicativo gamificado é quando se obtém um conteúdo real e tangível, como a proposta deste trabalho (ensinar lógica de programação), e que pode ser passado ao usuário fora do ambiente gamificado. Assim sendo é adicionado elementos de jogos, como, demonstração de resultados, estrelas/medalhas pelo cumprimento de um exercício e etc. Um caso desse é o Duolingo que foi apresentado anteriormente.

A seguir, será realizada a comparação entre este e os apresentados acima. Segue tabela de comparação entre os projetos.

Tabela 9.1 – Tabela comparação entre trabalhos relacionados e este projeto

|  |  |  |  |
| --- | --- | --- | --- |
| Trabalho | Interface | Motivação | Tipo de aprendizado |
| Minecraft | 3 áreas | Lúdica | Blocos de comando lúdicos |
| CodeCombat | 3 áreas | Lúdica | Linhas de comando lúdicas |
| HistoCode | 3 áreas | Real | Blocos de comandos ajustados para pensamento lógico computacional |

Como pode ser visto na tabela acima a interface de todos os 3 projetos são muito parecidas, possuindo 3 áreas principais, uma sendo a visualização do personagem e seus movimentos, uma com os comandos disponíveis para o desafio e a última sendo a área na qual o usuário utiliza os comandos disponíveis para resolver o desafio.

A Motivação para o usuário nos dois trabalhos relacionados são lúdicas, ou seja, não é algo real, no primeiro caso é sobre o famoso jogo Minecraft, no segundo caso é um jogo inventado pelo criador, no qual o usuário deve seguir por um labirinto completando objetivos e evitando inimigos. No caso deste projeto a motivação é real, pois utiliza a história da computação como motivação.

O tipo de aprendizado apresentado pelo jogo Minecraft é utilizando blocos de comando lúdicos, ou seja, os comandos representam ações dentro do jogo, como Plantar, Colher, Tosar. No caso do CodeCombat, as linhas de comando também são lúdicas, ou seja, podem ser linhas de comando parecidas com uma linguagem de programação real, porém não o é, e sim comandos como “herói mova-se para direita”, não representando algo concreto na área de programação. No projeto apresentado aqui, utilizou-se blocos de comandos pensando em lógica computacional, utilizando-se disso em sala de aula, futuramente quando a aula chegar ao ponto de ensinar programação, todos os desafios propostos podem ser transferidos facilmente para qualquer linguagem de programação.

O diferencial do HistoCode com relação aos outros trabalhos é a capacidade de desenvolver o aluno em uma linguagem mais próxima da utilizada para a programação em código, não utilizando-se do lúdico, pois o mesmo pode gerar dúvidas quando o usuário começar a utilizar uma linguagem de programação propriamente dita.

Além disso, todos os desafios podem ser facilmente traduzidos para muitas linguagens de programação, fazendo assim, com que o usuário possa representar novamente os desafios em uma linguagem propriamente dita. Com relação a motivação do HistoCode, ela é real, passando um conteúdo que o usuário pode aprender enquanto utiliza o mesmo, não sendo algo que o usuário só precisa visualizar, como uma ovelha ou árvore.

A seguir serão apresentados os trabalhos futuros com relação ao projeto.

## Trabalhos Futuros

Este projeto ainda possui muitos itens a serem revistos para a plena utilização em sala de aula, como a revisão e melhoria dos objetivos dos desafios, pois muito usuários tiveram dificuldades com os mesmos. Ainda é necessário ser criada uma maneira de melhorar a explicação do funcionamento dos comandos para o usuário, pois muitos comandos estavam mal explicados ou não, mesmo com a explicação provida, não era o suficiente para o completo entendimento do mesmo. Com relação a História da Computação tem de se rever como utilizá-la, pois a mesma não está bem integrada a maneira que foi realizado o aplicativo, não possuindo uma real integração com o desafio e sim sendo utilizada somente como um meio de criar o mesmo.

No geral, podemos considerar que o aplicativo atingiu o seu objetivo que era desenvolver um aplicativo gamificado que auxilie no desenvolvimento do raciocínio lógico e computacional usando como temática a história da computação, porém faltaram certos elementos, como maiores informações sobre os comandos e melhor explicação sobre os desafios, não atingindo satisfatoriamente os usuários pesquisados. Mesmo assim, muitos usuários consideraram que aprenderam mais sobre a história da computação e que esta ferramenta, ou como esta, poderia ser utilizada como auxilio no ensino. Com mais tempo de desenvolvimento e novas funcionalidades, pode tornar-se uma ferramenta presente em sala de aula, sendo utilizada como auxiliador no ensino em componentes curriculares, principalmente iniciais, de cursos que necessitam o desenvolvimento do pensamento lógico e computacional do aluno.

Apêndice

APÊNDICE A – Pesquisa com o usuário



APÊNDICE B – Pesquisa observacional



Referências Bibliográficas

AGUILERA, M. e MÉNDIZ, A. “Video Games and Education (Education in the Face of “Parallel School”)”. 2000. Disponível em: <http://dl.acm.org/citation.cfm?id=950583>. Último acesso: 05 jun 2017.

FARDO, Marcelo. A Gamificação Aplicada Em Ambientes De Aprendizagem. 2013. Disponível em: <http://seer.ufrgs.br/renote/article/viewFile/41629/26409>. Último acesso: 05 jun 2017.

JESUS, Alexandre Negrão et al. Objetos De Aprendizagem No Ensino De Lógica De Programação. 2007. Disponível em: < http://www.ria.net.br/index.php/ria/article/viewFile/23/23 >. Último acesso: 05 jun 2017.

LACERDA, D. et al.“Design Science Research: método de pesquisa para a engenharia de produção”. 2013. Disponível em: <http://www.scielo.br/pdf/gp/v20n4/aop\_gp031412.pdf>.

MOUSQUER, Tatiana, ROLIM, Carlos Oberdan. A Utilização De Dispositivos móveis Como Ferramenta Pedagógica Colaborativa Na Educação Infantil. 2011. Universidade Regional Integrada das Missões e do Alto Uruguai – URI, Santo Ângelo, RS - Brasil. Disponível em: < http://www.santoangelo.uri.br/stin/Stin/trabalhos/11.pdf>. Último acesso: 05 jun 2017.

MUNTEAN, Cristina Ioana. “Raising engagement in e-learning through gamification”. 2011. Disponível em: <http://icvl.eu/2011/disc/icvl/documente/pdf/met/ICVL\_ModelsAndMethodologies\_paper42.pdf>. Último acesso: 05 jun 2017.

NEVES, E., BORUCHOVITCH, E. “A Motivação de Alunos no Contexto da Progressão Continuada”. 2004. Disponível em: <http://www.scielo.br/pdf/%0D/ptp/v20n1/a10v20n1.pdf>.

OLIVEIRA, Millena Elena et al. Ensino de lógica de programação no ensino fundamental utilizando o Scratch: um relato de experiência. 2014. Disponível em: < http://www.lbd.dcc.ufmg.br/colecoes/wei/2014/0022.pdf >. Último acesso: 05 jun 2017.

PRENSKY, Marc. Digital Natives, Digital Immigrants. MCB University Press, 2001. Disponível em: < http://www.marcprensky.com/writing/Prensky%20-%20Digital%20Natives,%20Digital%20Immigrants%20-%20Part1.pdf>. Último acesso: 05 jun 2017.

PRODANOV, Cleber Cristiano; FREITAS, Ernani Cesar de. Metodologia do trabalho científico: métodos e técnicas da pesquisa e do trabalho acadêmico. 2. ed. Novo Hamburgo – RGS: Universidade Feevale, 2013. Disponível em: < http://www.feevale.br/Comum/midias/8807f05a-14d0-4d5b-b1ad-1538f3aef538/E-book%20Metodologia%20do%20Trabalho%20Cientifico.pdf>. Último acesso: 05 jun 2017.

RAPKIEWICZ, Clevi Elena et al. Estratégias Pedagógicas No Ensino De Algoritmos E Programação Associadas Ao Uso De Jogos Educacionais. 2006. Disponível em: < http://www.seer.ufrgs.br/renote/article/view/14284/0 >. Último acesso: 05 jun 2017.

SCAICO, Pasqueline Dantas et al. Implementação de um Jogo Sério para o Ensino de Programação para Alunos do Ensino Médio Baseado em m-learning. 2012. Disponível em: <http://www.imago.ufpr.br/csbc2012/anais\_csbc/eventos/wei/artigos/Implementacao%20de%20um%20Jogo%20Serio%20para%20o%20Ensino%20de%20Programacao%20para%20Alunos%20do%20Ensino%20Medio%20Baseado%20em%20mlearning.pdf > Último acesso: 05 jun 2017.

SCOLARI, A., BERNARDI G, CORDENONSI, A.. “O Desenvolvimento do Raciocínio Lógico através de Objetos de Aprendizagem”. 2007. Disponível em: <http://www.seer.ufrgs.br/renote/article/viewArticle/14253>. Último acesso: 05 jun 2017.

SOMBRIO, G., HAEMING, W., ULBRITCH, V. “Aprendizagem Criativa na Educação Utilizando Jogos e Gamificação”. 2014. Disponível em: <http://www.latec.ufrj.br/revistas/index.php?journal=hipertexto&page=article&op=view&path%5B%5D=642>. Último acesso: 05 jun 2017.

SOUZA, Natalia Galvão Simão de. O ensino da lógica na educação básica. 2013. 91 f. Trabalho de conclusão de curso (Licenciatura - Matemática) - Universidade Estadual Paulista, Faculdade de Engenharia de Guaratinguetá, 2013. Disponível em: <http://hdl.handle.net/11449/121446>. Último acesso: 05 jun 2017.

SQUIRE, K. “Changing the Game: What Happens When Video Games Enter the Classroom?”. 2005. Disponível em: <http://website.education.wisc.edu/~kdsquire/tenure-files/26-innovate.pdf>. Último acesso: 05 jun 2017.

TAROUCO, Liane et al. Jogos Educacionais. 2004. Disponível em: <http://www.virtual.ufc.br/cursouca/modulo\_3/Jogos\_Educacionais.pdf>. Último acesso: 05 jun 2017.

VIANNA, Maurício et al. Gamification, Inc., 2013. Disponível em: <http://www.livrogamification.com.br/>. Último acesso: 05 jun 2017.

ZICHERMANN, G., CUNNIGHAM, C. “Gamification by Design”. 2011. Disponível em: <http://66.45.248.24/books/[2011]%20Gamification%20by%20Design.pdf>. Último acesso: 05 jun 2017.

1. https://pt.wikipedia.org/wiki/Aplicativo\_m%C3%B3vel [↑](#footnote-ref-1)
2. https://pt.wikipedia.org/wiki/Jogo [↑](#footnote-ref-2)
3. https://dicionariodoaurelio.com/jogo [↑](#footnote-ref-3)
4. http://portal.inep.gov.br/web/portal-ideb/planilhas-para-download [↑](#footnote-ref-4)
5. https://code.org/ [↑](#footnote-ref-5)
6. https://scratch.mit.edu/ [↑](#footnote-ref-6)
7. https://dicionariodoaurelio.com/jogo [↑](#footnote-ref-7)
8. https://pt.wikipedia.org/wiki/Jogo [↑](#footnote-ref-8)
9. https://code.org/ [↑](#footnote-ref-9)
10. https://pt.duolingo.com/ [↑](#footnote-ref-10)
11. http://appinventor.mit.edu/explore/ [↑](#footnote-ref-11)
12. https://developer.mozilla.org/pt-BR/docs/Web/HTML [↑](#footnote-ref-12)
13. https://developer.mozilla.org/pt-BR/docs/Web/CSS [↑](#footnote-ref-13)
14. https://pt.wikipedia.org/wiki/JavaScript [↑](#footnote-ref-14)
15. https://pt.wikipedia.org/wiki/WebGL [↑](#footnote-ref-15)
16. https://pt.wikipedia.org/wiki/Unity [↑](#footnote-ref-16)
17. https://www.typeform.com/ [↑](#footnote-ref-17)
18. https://code.org/ [↑](#footnote-ref-18)