

**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E
TECNOLOGIA DE SÃO PAULO
CÂMPUS CAMPINAS**

**PAULO HENRIQUE BELUCCI
SILVIO DA COL DE BRITO**

R.A. 3001369

R.A. 300130X

**JOGO WEB PARA ENSINO DE ALGORITMO DESTINADO AO
PÚBLICO INFANTO-JUVENIL**

Campinas
2019

**PAULO HENRIQUE BELUCCI
SILVIO DA COL DE BRITO
R.A. 3001369
R.A. 300130X**

**JOGO WEB PARA ENSINO DE ALGORITMO DESTINADO AO
PÚBLICO INFANTO-JUVENIL**

Projeto de Pesquisa apresentado como requisito parcial para aprovação na disciplina “Metodologia de Pesquisa Tecnológica” do curso de Tecnologia em Análise e Desenvolvimento de Sistemas do Câmpus Campinas.

Campinas
2019

RESUMO

Este trabalho de pesquisa objetiva apresentar o desenvolvimento de um jogo como ferramenta de apoio ao ensino de algoritmo ao público infanto-juvenil. Dada a crescente demanda por profissionais de TI no mercado de trabalho frente a um déficit na formação profissionais qualificados, torna-se necessário o reforço do ensino do pensamento lógico e estruturado já durante a educação básica, de forma a preparar as crianças para as constantes mudanças no mercado de trabalho. Para tanto, o jogo resultante desta pesquisa poderá ser utilizado como uma forma de introduzir tal disciplina de maneira lúdica e motivadora, utilizando-se de recursos visuais atrativos além da temática de gamificação. O jogo poderá ser acessado de qualquer plataforma, e online, podendo ser utilizado em salas de aula ou em casa, e abordará tópicos fundamentais de qualquer linguagem de programação, como: comandos sequenciais, estruturas de decisão e de repetição.

Palavras-chave: Jogos digitais; Educação; Algoritmo; Gamificação.

LISTA DE FIGURAS

FIGURA 1	Caso de uso.....	10
FIGURA 2	Arquitetura do sistema.....	12
FIGURA 3	Plano de fundo para o jogo.....	13
FIGURA 4	Personagens.....	13
FIGURA 5	Sprite Sheet.....	14
FIGURA 6	Opções apresentadas no menu de ações.....	14
FIGURA 7	Exemplo de desafio proposto e interface que o jogador encontrará dentro do jogo.....	15

LISTA DE TABELAS

TABELA 1	Atividades do modelo clássico.....	10
TABELA 2	Cronograma sugerido.....	16

SUMÁRIO

1. INTRODUÇÃO.....	5
2. OBJETIVOS.....	6
2.1 OBJETIVO GERAL.....	6
2.2 OBJETIVOS ESPECÍFICOS.....	6
3. JUSTIFICATIVA.....	7
4. FUNDAMENTAÇÃO TEÓRICA.....	8
5. METODOLOGIA.....	10
6. CRONOGRAMA.....	16
7. REFERÊNCIAS.....	17

1. INTRODUÇÃO

O mercado de trabalho atualmente possui uma grande necessidade de profissionais na área de tecnologia com diferentes habilidades tais como, design, linguagem de programação, gestão, desenvolvimento de soluções e outras habilidades inerentes à área de TI. Segundo o relatório da Associação Brasileira das Empresas de Tecnologia da Informação e Comunicação (Brasscom, 2019) 420 mil profissionais serão demandados entre 2018 e 2024, ou seja, aproximadamente 70 mil profissionais por ano. Ainda de acordo o mesmo relatório no Brasil forma-se apenas 46 mil profissionais com perfil tecnológico, resultando em uma diferença de aproximadamente 24 mil profissionais ao ano. Esse comportamento se repete em todo o mundo e verifica-se uma necessidade urgente de formar profissionais nesta área.

Essa falta de profissionais tem raízes profundas na deficiência que a educação básica possui. Apesar das mudanças pelas quais o mundo passa atualmente, relacionadas ao uso das tecnologias e principalmente das linguagens de programação, não há um esforço por parte dos governos no sentido de adequar o ensino, e preparar as crianças para a realidade que o mundo e o mercado necessitam.

Nesse contexto o assunto programação e por vezes a informática em geral acaba sendo introduzida no aprendizado tardiamente, o que faz com que o assunto não seja de interesse dos alunos e por consequência, como dito anteriormente faltem pessoas interessadas em trabalhar nesse mercado de trabalho.

Com o intuito de preencher essa lacuna educacional, diversas iniciativas oferecem o ensino de linguagem de programação através de jogos digitais como o Scratch (MIT, 2007), code Monkey (Schor, 2014), Lightbot (Lightbot Inc., 2008) e outros jogos que tem como objetivo introduzir os conceitos iniciais de linguagem de programação e fomentar o interesse de crianças e jovens pelo assunto, já que o tema faz parte do cotidiano desses jovens sempre conectados, mesmo que de forma implícita.

Desta forma, este projeto propõe apresentar um jogo que irá auxiliar o ensino de programação, buscando de forma lúdica introduzir conceitos básicos para desenvolvimento de código.

1. OBJETIVOS

1.1 OBJETIVO GERAL

Desenvolver um jogo que auxilie no ensino de linguagem de programação para o público infanto-juvenil, abordando inicialmente o tema de algoritmos e linguagem estruturada.

1.2 OBJETIVOS ESPECÍFICOS

- Pesquisar métodos didáticos aplicados ao ensino de linguagem de programação;
- Levantar ferramentas para desenvolvimento de jogos voltados à plataforma WEB;
- Modelar e desenvolver o jogo por meio das ferramentas pesquisadas aplicando um método previamente selecionado;
- Disponibilizar o jogo através da internet para aplicação de testes e validação de resultados.

3. JUSTIFICATIVA

Cursos de programação geralmente apresentam altas taxas de reprovação e, até mesmo, desistência, pois o ensino desta área de conhecimento é considerado uma tarefa complexa. (Piteira e Samir, 2011)

Experiência prévia com programação mostrou impacto positivo em alunos de universidades, matriculados em cursos relacionados à ciência da computação de acordo com Grandell (2006). No entanto, ainda segundo Grandell, tal aprendizado continuará a beneficiar mesmo alunos que não pretendem seguir a área de tecnologia, pois contribui para o desenvolvimento de meta habilidades como resolução de problemas, pensamento estruturado e lógico.

Em um estudo sobre a relação entre o avanço tecnológico e seu impacto no mercado de trabalho no Brasil, entre 2000 a 2015, Cintra (2018) aponta que a tecnologia pode interferir de duas formas, eliminando antigas profissões e criando novas. O que indica a importância de habituar-se a novas tecnologias e a seu funcionamento para a manutenção da empregabilidade.

4. FUNDAMENTAÇÃO TEÓRICA

Desde os tempos antigos o jogo é utilizado como uma forma de amenizar ou até mesmo substituir de forma figurada, a realidade em que vivemos, McGonigal (2011). De acordo com HUIZINGA (2005), o jogo é uma atividade voluntária, portanto qualquer movimento no sentido de ordenar esta atividade ao praticante, deixa de ser um jogo. Ainda segundo HUIZINGA, uma notória característica que define o que é o jogo, é o fato deste evadir a realidade, geralmente levando o jogador para um contexto diferente daquele em que ele está estabelecido.

Já de acordo com KAPP (2012), jogo é “um sistema em que os jogadores se engajam em um desafio abstrato, definido por regras, interatividade e feedback, que resulta em uma saída quantificável e frequentemente provoca uma reação emocional”.

Utilizando como base essas duas definições citadas acima podemos entender que o jogo é uma forma de o ser humano transformar o seu entendimento sobre determinada atividade, buscando diversão, prazer e até mesmo significado em suas ações.

Para BATTAIOLA (2000) três partes básicas formam um jogo:

- ✓ Enredo (Tema, trama e objetivos);
- ✓ Interface Interativa (Representação gráfica dos estados de um jogo);
- ✓ Motor de jogo (Mecanismo que controla a reação do jogo às ações do jogador)

O motor do jogo (MJ) tem um papel importante em oferecer ferramentas prontas para o desenvolvimento do mesmo e deve ser escolhido em função da linguagem e ambiente em que o software virá a ser executado.

EBERLY (2001) afirma que o MJ engloba bibliotecas para manipulação de recursos fundamentais para um jogo, como gráficos, simulação de fenômenos físicos, inteligência artificial, funcionamento em rede e codificação.

Além do que o jogo representa, e suas características, abordaremos neste projeto um aspecto fundamental para o engajamento dos usuários, a gamificação.

Segundo KAPP (2012) a gamificação é “o uso de mecânicas, estéticas e pensamentos dos games para engajar pessoas, motivar a ação, promover a

aprendizagem e resolver problemas” e é esse o ponto de partida para a criação do jogo, tornar o aprendizado algo que permeia a definição encontrada por KAPP, onde a motivação se dará não por meio de absorção de conhecimento pura e simplesmente, mas sim pelo fato de o jogo, aliado a gamificação, tornar o processo de aprendizagem divertido.

Aplicações Web têm crescido rapidamente tanto em termos de uso, quanto em seu escopo, impactando diversos aspectos de nossas vidas (Ginige e Murugesan, 2001).

Para reduzir significativamente o tempo de desenvolvimento, equipes de desenvolvimento utilizam-se de *frameworks*. Um framework é uma aplicação semi-completa, a qual pode ser reutilizada e customizada para produzir aplicações completas (Johnson, 1997).

Um padrão arquitetural muito utilizado na construção de diversos frameworks Web é o MVC (LEMOS, 2013), embora seja criado antes mesmo do advento da internet pelos criadores da linguagem Smalltalk na década de 80 (Krasner, 1988).

Outro aspecto essencial à produção de software é a definição de uma metodologia de desenvolvimento. Nota-se no mercado atualmente duas vertentes, sendo uma a de métodos tradicionais versus a de métodos ágeis. Metodologias tradicionais são mais adequadas a processos que disponham de requisitos estáveis e previsíveis, e também é conhecida por ser pesada e orientada a planejamento (SOARES, 2014). Em situações onde ocorrem mudanças constantes, as equipes são pequenas e o alto custo de alterações em software se torna restritivo, o desenvolvimento rápido se torna uma alternativa às metodologias ágeis, ainda segundo SOARES.

5. METODOLOGIA

A modelo de desenvolvimento adotado para o jogo será o clássico, também conhecido como sequencial (PRESSMAN, 2001). Este mostra-se adequado, uma vez que o projeto tem seus requisitos bem conhecidos e não exigirá grandes mudanças em sua estrutura. Um processo de software consiste de etapas que, associadas, possibilitam a produção de um software (SOMMERVILLE, 2003). As atividades que compõe o modelo clássico são apresentadas, em sequência de execução, na tabela 1:

Tabela 1 - Atividades do modelo clássico

Ordem	Descrição
1	Definição de requisitos
2	Projeto de software
3	Implementação e teste de unidades
4	Integração e teste de sistema
5	Operação e manutenção

O jogo oferecerá 3 fases, onde cada fase exigirá do jogador a aplicação de um novo conceito, ou habilidade. O diagrama de caso de usos é apresentado na figura 1 e exibe as possíveis interações com o usuário:

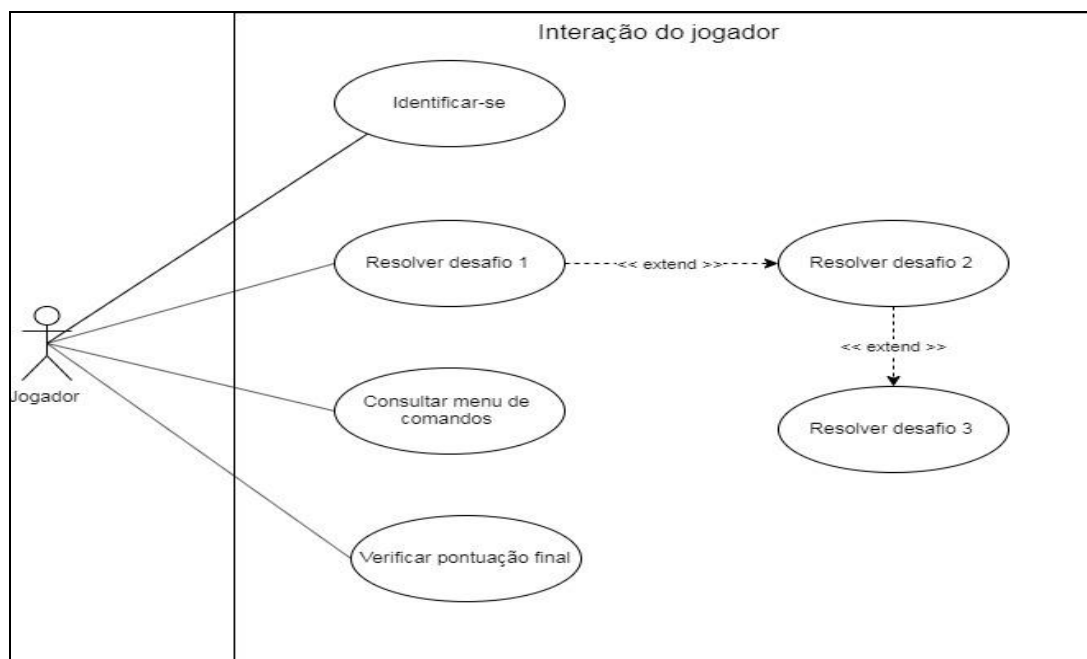


Figura 1 - Caso de uso

O jogo será desenvolvido para plataformas web, de forma a atender ao requisito de ser multiplataforma, ou seja, permite-se ser executado em qualquer sistema operacional, necessitando apenas de um navegador.

Para o desenvolvimento serão utilizadas as três principais ferramentas adotadas atualmente na confecção de conteúdo para o ambiente web, quais sejam, HTML5 (linguagem de marcação), CSS (estilização), JavaScript (Linguagem de programação).

O HTML5 permite criar a estrutura do jogo, delimitando o espaço que cada componente irá utilizar dentro da tela em que será apresentado. Um dos principais recursos do HTML5 e que será utilizado para o desenvolvimento deste projeto é o Canvas (tela de pintura), que representa basicamente o espaço e os limites em que os elementos do jogo serão dispostos e manipulados (Meyer, 2011) através de outras tecnologias como CSS e JavaScript.

Já para a estilização, animação e outros aspectos visuais do jogo será utilizado o recurso de Folhas de Estilo em Cascata (CSS), do inglês *Cascading Style Sheets*.

O CSS é um mecanismo para adicionar estilo (cores, fontes, espaçamento, etc.) a um documento web (W3C, 2011), também pode aplicar animações e definir o comportamento, visualmente falando, dos componentes dispostos no Canvas.

Já para a parte da lógica do jogo, a qual incluirá, animações, operações matemáticas, sistema de recompensas, entre outros, será utilizada a linguagem de programação JavaScript.

O Javascript é uma linguagem de programação leve, interpretada pelos *browsers* (navegadores web). Por meio desta linguagem podem ser manipulados quaisquer elementos de um documento HTML, definindo estilos, mudando suas propriedades, definindo ações, entre outros aspectos (FLANAGAN, 2004).

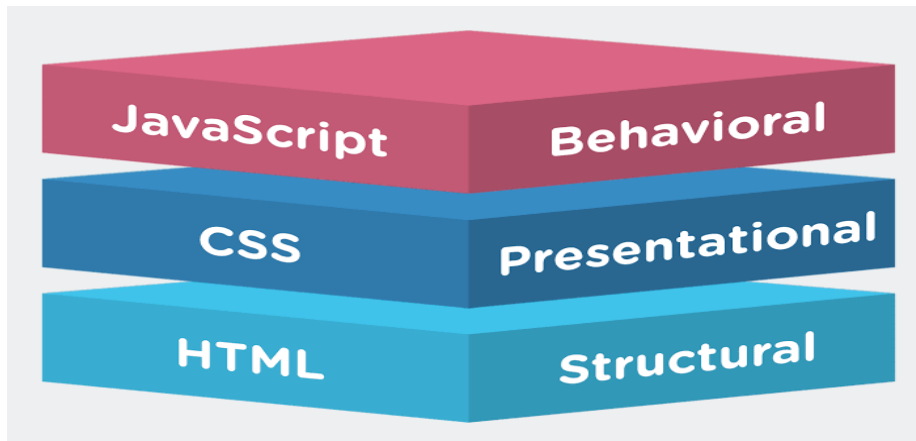


Figura 2 - Arquitetura do sistema

Fonte: <http://www.cagrimmett.com/development/2017/04/24/tools-for-learning-css.html>

Com relação a ao design e visualização do jogo. Serão adotados temas e jogabilidade de acordo com o público alvo do projeto que serão crianças.

Para HUIZINGA, o fator lúdico é fundamental no engajamento do jogador com o jogo, pois transmite ao jogador uma nova dimensão e que distorce a realidade ou cria uma nova, fazendo com o que o jogo tenha um caráter prazeroso e satisfatório a medida em que os desafios propostos são atingidos. Desta forma o jogo será composto por uma temática infantil, abordando elementos e personagens que remetem a algo caricato.

Abaixo temos um exemplo do plano de fundo (*background*) utilizado para o desenvolvimento do jogo, bem como a apresentação dos personagens que irão compor o jogo.

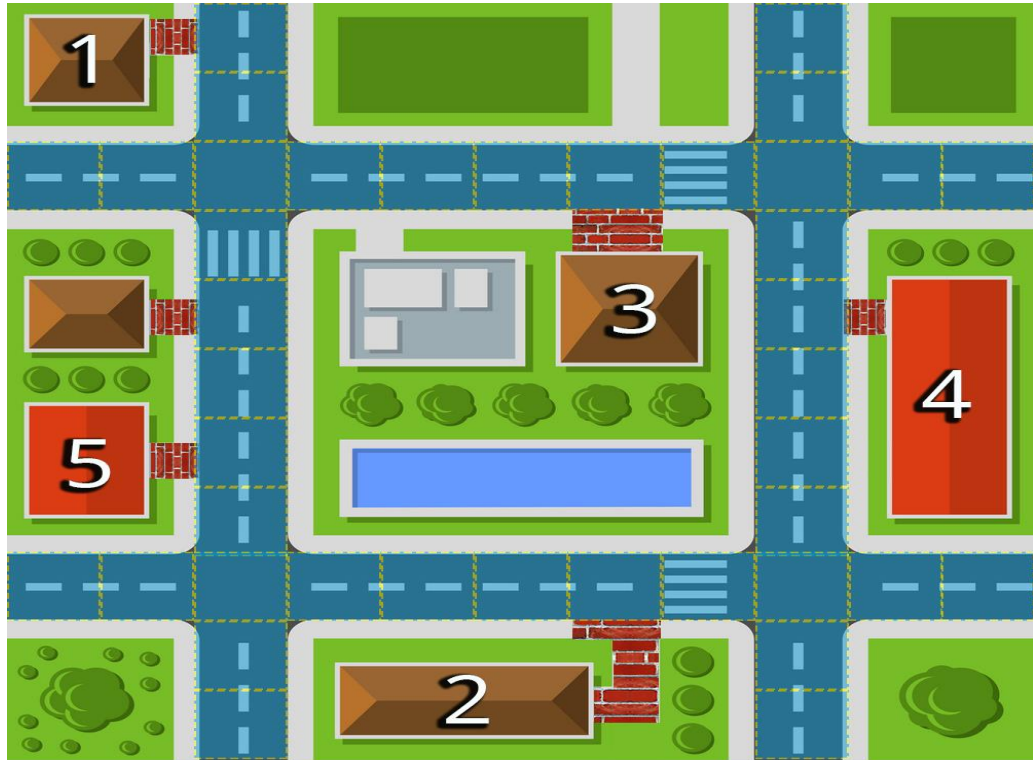


Figura 3 – Plano de fundo para o jogo.



Figura 4 – Personagens

O processo de animação dos personagens e demais componentes posicionados na tela de pintura (canvas/HTML) será feita através de scripts¹ utilizando a linguagem de programação JavaScript. Para realizar a animação os scripts possuem como base uma série de imagens chamadas Sprites² que vão sendo substituídas por outras semelhantes, porém em estados diferentes a cada frame³.

1 - Scripts são um conjunto de instruções para que uma função seja executada em determinado aplicativo.

2 - Em computação gráfica, um sprite é um objeto gráfico bi ou tridimensional que se move numa tela sem deixar traços de sua passagem.

3 - Quadro de vídeo, também conhecido como frames de vídeo ou frames por segundo, é cada uma das imagens fixas de um produto audiovisual.

Na figura 5 pode ser conferido um exemplo de Sprite sheet, uma sequência de sprites utilizados em uma animação.



Figura 5 – Sprite Sheet.

Como dito anteriormente o jogo terá 3 fases que irão envolver desafios simples, porém que apresentem alguma complexidade logica para sua resolução. Os desafios consistirão em controlar o personagem com o objetivo de pegar uma chave em determinado ponto do mapa e abrir uma porta em outro ponto, porém utilizando scripts para a realização da tarefa. Os comandos que o usuário poderá utilizar para escrever o script será disponibilizado através de um menu que poderá ser acessado a qualquer momento do jogo.

Abaixo temos um exemplo de como o menu será apresentado e as informações contidas neste elemento.

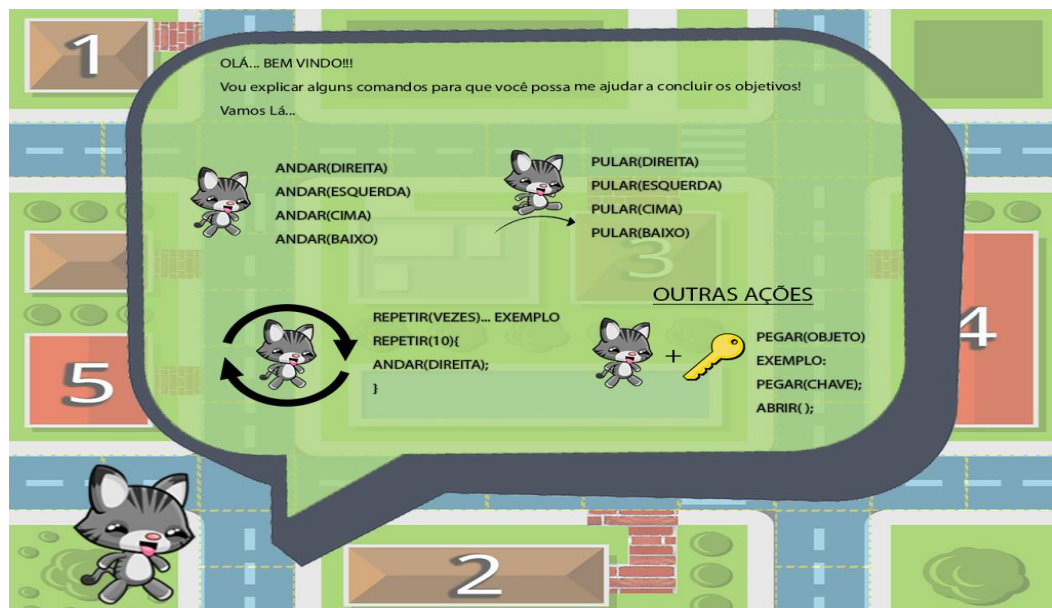


Figura 6 – Opções apresentadas no menu de ações

Os comandos fornecidos poderão ser digitados em uma área delimitada à direita do mapa, onde as linhas poderão ser adicionadas e ao final da inserção dos comandos desejados a instrução poderá ser executada. Uma parte importante para o funcionamento do jogo é que o script seja executado em sua totalidade para concluir os desafios, ou seja, o jogador deverá digitar todas as ações que o personagem precisa realizar para atingir o objetivo e só depois clicar no botão para executar o script.

Desta forma, o pensamento computacional será essencial para que os objetivos sejam atingidos.

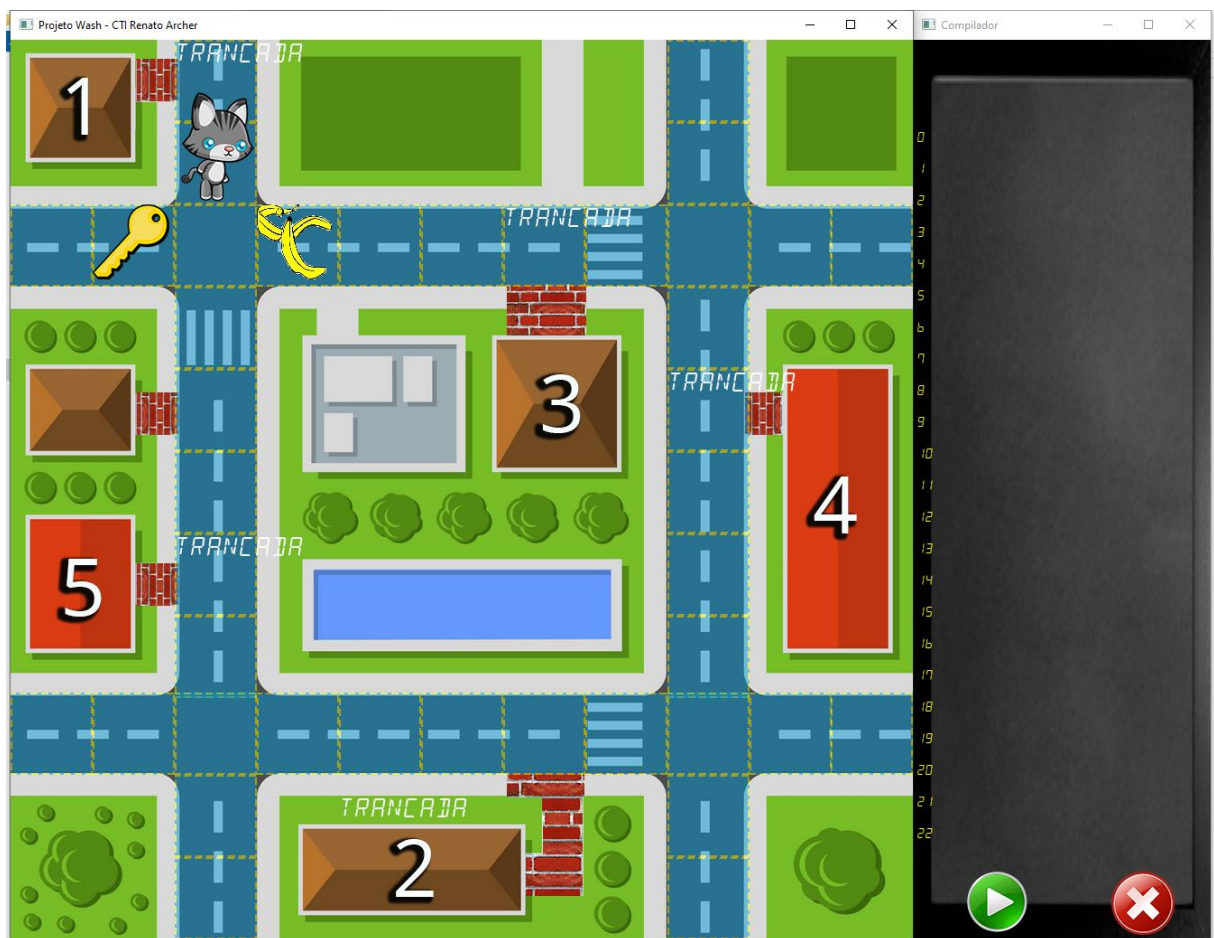


Figura 7 – Exemplo de desafio proposto e interface que o jogador encontrará dentro do jogo.

6. CRONOGRAMA

ATIVIDADES		Jul	Ag	Set	Out	Nov	Dez
1	Revisão Bibliográfica e Fundamentação Teórica.						
2	Elaboração de Projeto de Pesquisa						
3	Entrega de Projeto de Pesquisa						
4	Envio para análise do orientador						
5	Apresentação do Projeto escrito para análise						
6	Coleta de Dados para estruturação do projeto						
7	Compilação de Dados e/ou Informações Obtidas						
8	Avaliação dos Resultados						
9	Elaboração de Material						
10	Apresentação do projeto para Banca Examinadora						

7. REFERÊNCIAS

BATTAIOLA, A. L. (2000). Jogos por computador: Histórico, relevância tecnológica e mercadológica, tendências e técnicas de implementação. Anais do XIX Jornada de Atualização em Informática. Curitiba: SBC 2000.

Brasscom - Relatório Setorial de TIC (2019). Disponível em: <https://brasscom.org.br/relatorio-setorial-de-tic-2019>. Acesso em: 15 de Outubro de 2019.

Cintra, Luciano Mendes (2018). A relação entre o progresso tecnológico e o desemprego no Brasil nos anos de 2000 a 2015. Franca: UniFacef, 2018.

EBERLY, D. H. (2001). 3D game engine design: a practical approach to real-time computer graphics. São Francisco: Morgan Kaufmann.

FLANAGAN, David. JavaScript: o guia definitivo. Porto Alegre: Bookman, 2004;
SILVA, Mauricio Samy. Criando Sites com HTML: Sites de Alta Qualidade com HTML e CSS. São Paulo: Novatec, 2008.

Ginige, A., Murugesan, S. (2001), "Web Engineering: na Introduction", IEEE Multimedia, Vol. 8, Issue: 1, pp: 14 – 18.

Grandell, L.; Peltomaki, M.; Back, R. B.; Salakoski, T. (2006). Why Complicate Things? Introducing Programming in High School Using Python. Anais do 8th Australasian Computing Education Conference (ACE 2006). Hobart, Tasmania, Australia.

HUIZINGA, J. Homo ludens. São Paulo: Perspectiva, 2005.

Johnson, R. E. (1997). "Frameworks, Components, Patterns", Communications of the ACM, v.40, n.10, pp. 39-42, 1997

Kapp, K. M. (2012). The Gamification of Learning and Instruction: Game-based Methods and Strategies for Training and Education

Krasner (1988), S. T. Pope, "A cookbook for using the model-view controller user interface paradigm in Smalltalk80", Journal of Object-Oriented Programming, v.1 n.3, Aug./Sept. 1988, pp.26-49.

LEMONS, M. F.; OLIVEIRA, P. C.; RUELA, L. C.; SANTOS, M. S. (2013). APLICABILIDADE DA ARQUITETURA MVC EM UMA APLICAÇÃO WEB(WebApps). Revista Eletrônica Científica de Ciência da Computação.

McGonigal, J. (2011). Reality is broken: Why games make us better and how they can change the world. Penguin Group, The.

MEYER, J. (2011). O Guia Essencial do HTML5: Usando jogos para aprender HTML5 e Javascript. Rio de Janeiro: Ciência Moderna.

MIT (2007): <https://scratch.mit.edu>. Acessado em 10 de Outubro de 2019.

Piteira, Martinha & Haddad, Samir. (2011). Innovate in your program computer class: An approach based on a serious game. 10.1145/2016716.2016730.

Pressman, R. Engenharia de Software. McGraw-Hill, 2001.

SCHOR, Jonathan (2014): <https://www.codemonkey.com/about-us>. Acessado em 05 de Outubro de 2019.

SOARES, M. D. SANTOS (2014). Comparação entre Metodologias Ágeis e Tradicionais para o Desenvolvimento de Software. 68-1-108-1-10-20140918.

Sommerville, I. Engenharia de Software. Editora Addison-Wesley. 592p, 2003

W3C BRASIL. Conhecendo o W3C. São Paulo, WC3, 2011. Disponível em:

<<http://www.w3c.br/Sobre/ConhecendoW3C>>. Acesso em 20 de Setembro de 2014.