

# Automigos: *learning design* para ludificação de Autômatos Finitos Determinísticos

Fabrizio Honda<sup>1</sup>, Fernanda Pires<sup>1</sup>, Marcela Pessoa<sup>1</sup>, Elaine H. T. Oliveira<sup>2</sup>

<sup>1</sup> Escola Superior de Tecnologia – Universidade do Estado do Amazonas (EST/UEA)  
ThinkTEd Lab - Pesquisa e Desenvolvimento em Tecnologias Educacionais

<sup>2</sup>Programa de Pós-Graduação em Informática  
Instituto de Computação – Universidade Federal do Amazonas (IComp/UFAM)

{fhf.lic17, fpires, mspessoa}@uea.edu.br, elaine@icomp.ufam.edu.br

**Abstract.** *Computing contents are considered complex and require a certain level of abstraction to understand them. One of the alternatives to facilitate its understanding is through educational games, which have been used to help in the learning of different contents. Thus, the present work aims to ludify a complex computing topic through a sequence of steps that result in an educational game. The artifact, entitled Automigos, takes into account the Constructivist learning theory, by Jean Piaget, and Computational Thinking, whose evaluation by the target audience was positive, regarding the aesthetics, mechanics and fun of the game in general, however, with reservations, mainly regarding the understanding of the rules and content.*

**Resumo.** *Os conteúdos de computação são considerados complexos e exigem um certo nível de abstração para compreendê-los. Uma das alternativas para facilitar sua compreensão é através dos jogos educacionais, que vêm sendo utilizados para auxiliar na aprendizagem de diversos conteúdos. Dessa forma, o presente trabalho tem o objetivo de ludificar um tema complexo de computação através de uma sequência de passos que resultam em um jogo educacional. O artefato, intitulado Automigos, leva em consideração a teoria de aprendizagem Construtivista, de Jean Piaget, e o Pensamento Computacional, cuja avaliação pelo público-alvo foi positiva, no que se refere a estética, mecânica e diversão do jogo no geral, no entanto, com ressalvas, principalmente quanto à compreensão das regras e conteúdo.*

## 1. Introdução

No que diz respeito aos cursos de Computação no ensino superior, observam-se altas taxas de desistência e reprovação. Apesar da pesquisa de Bennedsen e Carpensen [2019] – com 161 universidades/faculdades – indicar uma diminuição na taxa de reprovação de 33% para 28%, a Educação em Computação ainda é um desafio.

Dentre os motivos que contribuem com essas taxas, pode-se citar a didática dos professores, dificuldade das avaliações, saúde mental, etc. [Neto 2021]. Ademais, nos cursos superiores de computação, as disciplinas possuem ementas com conteúdos complexos, por exemplo, linguagens regulares contém: gramáticas livres de contexto, teoria dos autômatos, decidibilidade, entre outros [Dognini and Raabe 2003]. Tais

conteúdos demandam certa capacidade de abstração e seu entendimento é necessário para conteúdos mais avançados [Honda et al. 2022b]. Visando facilitar sua compreensão, pesquisadores têm investigado a utilização de jogos educacionais no aprendizado desses conteúdos [Melo et al. 2019, Macena et al. 2022, Honda et al. 2022a], pela capacidade de motivação e engajamento que proporcionam, podendo manter o jogador imerso.

A concepção desses jogos, no entanto, não é trivial, e a ludificação de um conteúdo complexo de computação também é considerado um desafio, visto que se faz necessária a compreensão tanto do tema abordado, quanto de teorias de aprendizagem, para conectar o *game design* ao *design* de aprendizagem [Honda et al. 2020].

Este trabalho descreve o processo de *learning design* na ludificação do conteúdo de Autômatos Finitos Determinísticos (AFD) para contribuir com a área de Educação em Computação, cujo artefato construído foi um jogo educacional intitulado “Automigos”. O artigo estrutura-se da seguinte forma: na Seção 2, a fundamentação teórica e trabalhos relacionados; na Seção 3, o passo a passo para a ludificação do tema de AFD; na Seção 4, os resultados e; por fim, as considerações finais.

## 2. Jogos, *flow* e conteúdos de computação

Os jogos estão presentes desde os primórdios da humanidade – antes “brincadeiras”, e possuem a diversão como ponto-chave [Huizinga 2014]. Estes possibilitam que o jogador entre em um estado de imersão, denominado *flow*, através de características como: mecânicas, objetivos claros, resposta direta, entre outros [Piacentini 2011]. Estando ele focado para obter um item especial, derrotar um “chefão” difícil ou algo do gênero, aquilo se torna seu objetivo momentâneo, ao qual fará de tudo para obtê-lo. Assim, os jogos podem vir a proporcionar interação, engajamento [Pires et al. 2018] e motivação entre os estudantes durante a *gameplay*, características que se têm buscado para a aprendizagem.

Dessa forma, surgem indagações: i) se é possível obter esse mesmo *flow* ao aplicar um processo de ludificação com elementos de jogos a um tema complexo de computação, resultando em um jogo educacional e; ii) se o jogador consegue se divertir enquanto aprende um conteúdo curricular de computação. Tais questões vão depender de como a ludificação foi realizada, visto que requer uma série de etapas e não é um processo trivial.

Por exemplo, Honda et al. [2022] apresentaram uma sequência didática que contém três etapas: concepção, ludificação e avaliação, visando a modelagem de um tema complexo de computação de forma lúdica. O artefato produzido foi um conjunto de mecânicas de um jogo educacional que trata sobre percurso em grafos, cujo objetivo do jogador é entregar pizzas utilizando rotas menos custosas e escolhendo os entregadores corretos. Apesar dos resultados positivos e dos benefícios dos jogos educacionais, pôde-se constatar, através das discussões dos autores, que sua concepção para temas complexos de computação também é uma atividade complexa.

Em Macena, Pires and Pessoa [2020], as autoras apresentaram uma proposta lúdica intitulada “Operação Lovelace”, que consiste em um jogo educacional com o propósito de promover o desenvolvimento do Pensamento Computacional com crianças de 8 a 12 anos. Para isso, o jogo envolve conceitos de algoritmos e estrutura de dados, proporcionando uma mecânica de *drag and drop*. A história do jogo aborda duas cientistas, Sam e Sabrina, que construíram um robô para auxiliá-las no laboratório Lovelace, mas

suas baterias estão espalhadas, devendo o jogador encontrá-las antes que ele descarregue. As avaliações apontaram resultados promissores, mas com ressalvas.

Já o trabalho de Melo et al. [2020] consiste de um Mapeamento Sistemático da Literatura sobre criação de jogos educacionais, cuja busca foi realizada nas bases de dados *Scopus*, *IEEE*, *ACM* e *Web of Science*. O objetivo da pesquisa era identificar, no que tange os jogos educacionais, as metodologias de desenvolvimento, teorias de aprendizagem aplicadas e suas vantagens para a aprendizagem. Resultados apontaram que os jogos educacionais podem auxiliar na aprendizagem, contudo, não há uma padronização de criação destes objetos e muitos autores não abordam suas avaliações.

Como se observa nesses trabalhos, os jogos educacionais podem ser poderosos aliados aos processos de aprendizagem, contudo, a concepção desses objetos não é trivial e a parte de avaliação é fundamental. Nesse contexto, diferente dos trabalhos de Honda et al. [2022] e Macena, Pires e Pessoa [2020] que tratam, respectivamente, de temas de teoria dos grafos e algoritmos/estrutura de dados, o presente trabalho visa contribuir com a ludificação do tema de Autômatos Finitos Determinísticos, resultando em um jogo educacional. O trabalho inclui uma série de etapas, incluindo avaliação, cujos tópicos foram poucos explorados nos trabalhos avaliados pela RSL de Melo et al. [2020].

### 3. Aplicando elementos de jogos em um tema de Computação

Aplicar elementos de jogos em um tema de computação requer um passo a passo minucioso. Para isso, este trabalho seguirá as etapas definidas na sequência didática de Honda et al. [2022]: concepção, ludificação e avaliação (Figura 1).

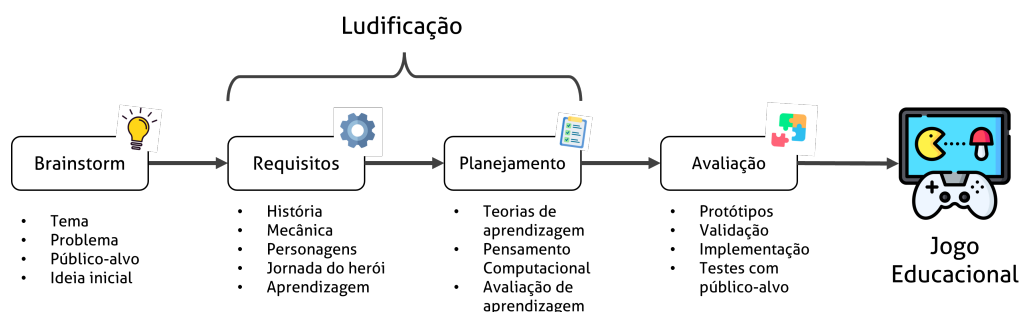


Figura 1. Etapas de elaboração de um artefato. Autor: Honda et al. [2022b]

#### 3.1. Concepção: *brainstorm*

O processo de concepção inicia-se com um *brainstorm* realizado pelo(s) autor(es), visando a elaboração das ideias iniciais do jogo. O primeiro passo é a definição de um problema de aprendizagem, que resulta também na definição do tema, conteúdo e público-alvo. O problema definido foi a dificuldade de aprendizagem em disciplinas que envolvem Linguagens Formais e Autômatos (LFA), visto que suas ementas contém conteúdos complexos e conceitos abstratos, como decidibilidade, gramáticas livre de contexto, autômatos, etc. [Dognini and Raabe 2003]. Como tema escolheu-se os Autômatos Finitos e o conteúdo foi Autômatos Finitos Determinísticos (AFD), ao qual o autor tem familiaridade. Portanto, o público-alvo definido foram estudantes de cursos superiores de computação, que já cursaram ou estão cursando disciplinas de LFA ou similares.

Os Autômatos Finitos são modelos matemáticos de máquinas computacionais – menos poderosos que uma Máquina de Turing – de estados finitos. Estes leem uma entrada (sequência de símbolos) de uma fita e reproduzem uma saída, indicando aceitação ou rejeição dessa entrada. O determinismo de um AFD caracteriza-se pela presença de um único estado inicial, cuja leitora da máquina avança para a posição seguinte da fita após um símbolo ser lido [Jukemura et al. 2005]. Existem muitas ferramentas que permitem a construção e visualização de AFDs para praticá-los, como é o caso do *JFLAP*<sup>1</sup>, *Language Emulator*<sup>2</sup>, e diversas outras ferramentas [Mioni and Barbosa 2022]. No entanto, apesar de auxiliarem com o exercício do conteúdo, essas ferramentas não possuem elementos lúdicos. Assim, algumas vezes, acabam não proporcionando motivação ao estudante.

Nesse intuito, com a possibilidade dos jogos educacionais para gerar motivação e engajamento, muitos jogos surgiram para auxiliar na aprendizagem de conteúdos de computação, dentre eles, os AFD. É, inclusive, o tema de LFA mais abordado em jogos educacionais [Santini et al. 2022]. No entanto, ainda é uma quantidade de jogos reduzida, quando comparada com os jogos educacionais presentes na Literatura e, quando o jogo foi concebido, eram poucos os que abordavam esse tema, sendo este um dos motivos que motivou o desenvolvimento do trabalho. Dessa forma, após a concepção inicial do *game*, a etapa seguinte descreve a ludificação de AFD.

## 3.2. Ludificação

A ludificação consiste na criação de elementos de jogos, que correspondem às etapas de requisitos e planejamento na construção de um jogo educacional. É necessário a documentação de todo esse processo, a qual se fez uso do *Educational Game Design Document*, uma documentação autoral de jogo educacional que considera o tripé: jogos, educação e processamento da informação.

### 3.2.1. História

Um dos primeiros elementos que é elaborado em um jogo educacional é a história, essencial para a criação dos demais elementos. Para isso, durante sua elaboração, o autor inicialmente optou por uma temática mais tropical envolvendo frutas – inspiração de um trabalho similar – e por personagens retangulares simpáticos que consumiriam essas frutas. No entanto, em discussão com outra autora, optou-se por uma temática de espaço/alienígena que possibilitaria criar uma narrativa mais interessante para o jogo.

Logo no *storytelling*, buscou-se apresentar referências ao tema que está sendo trabalhado: os Autômatos Finitos Determinísticos – i) o conceito de autômato no nome de um sistema dos personagens do jogo; ii) “Auto”, de autômatos, e “migos”, de amigos, na composição do nome desses personagens, denominados “Automigos”; e iii) o astronauta “Pitts” como personagem que conduz o tutorial, um cientista que, juntamente com McCulloch, foram os precursores do primeiro conceito similar ao de autômatos finitos [McCulloch and Pitts 1943]. Dessa forma, a história do jogo elaborada foi:

“Em vários planetas, numa galáxia distante, vivem os Automigos, seres muito inteligentes. Apesar de não serem muito amigáveis, são chamados dessa forma por serem

---

<sup>1</sup><https://www.jflap.org/>

<sup>2</sup><https://homepages.dcc.ufmg.br/lfvieira/ftc.html>



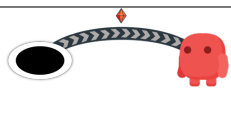


os criadores do Autômato, um sistema que distribui cristais para eles, seu alimento. Esses cristais também são utilizados como fonte de energia para o sistema, que são enviados diariamente para o planeta dos Automigos através da nave especial do comandante Pitts, encarregado de auxiliá-los. Porém, uma chuva de meteoros danificou os sistemas e Pitts teve de ir até o planeta terra para encontrar alguém que possa consertá-los.

Dessa forma, o jogador deve consertar os sistemas dos Automigos. Os cristais, fonte de alimento desses seres e energia dos sistemas, são gerados através de um tubo da nave de Pitts. Nesse sistema, há plataformas de energia e Automigos, que consomem esses cristais. Eles se conectam por meio de esteiras, que encaminham os cristais para outra plataforma ou para os Automigos, e são energizadas quando um cristal é consumido. Os cristais devem ser utilizados sequencialmente e o último deve ser consumido por um Automigo, sendo estas as regras do sistema.

### 3.2.2. Gameplay

O próximo elemento diz respeito à *gameplay* do jogo: a mecânica. Considerando que os AFD consistem de uma quintupla (5-upla), optou-se por manter essa configuração dentro do jogo, trazendo ludicidade a esses elementos (Tabela 1), de modo que: o alfabeto são os cristais; os estados são as plataformas de energia; a função de transição são as plataformas e Automigos conectados por esteiras; o estado inicial é a plataforma próxima ao tubo e; os estados de aceitação são os Automigos.

**Tabela 1. Elementos do jogo e correspondências em AFD.**

Elemento	Nome	Função
	Cristais	Alfabeto
	Plataformas de energia	Estados
	Esteiras que conectam plataformas e Automigos	Função de transição
	Plataforma próxima ao tubo	Estado inicial
	Automigos	Estados de aceitação

### 3.2.3. Mecânica de aprendizagem

Em sala de aula, nas disciplinas que incluem o conteúdo de AFD, é comum a utilização de papel e caneta para construir os autômatos. Já em formato digital, uma das ferramentas bastante utilizadas é o *JFLAP*, no qual o usuário consegue criar seus estados, estabelecer ligações e definir estados iniciais e finais. Considerando essa liberdade que o *software* possibilita ao usuário na criação de seus autômatos e por ser digital, optou-se por utilizar uma mecânica baseada no *JFLAP*. Ademais, visto que o mesmo só pode ser instalado em

computadores, definiu-se que o jogo seria destinado a dispositivos móveis, abrangendo um público maior.

Dessa forma, definiu-se que o objetivo do jogador é consertar o sistema dos Automigos, de acordo com a combinação de cristais enviadas àquele planeta. Em termos computacionais, o jogador terá de construir um AFD que aceite a entrada especificada. Para alcançar esse objetivo, o jogador deverá realizar ações com os elementos do jogo: posicionar plataformas e Automigos nos locais de encaixe, inserir esteiras e trocar os cristais de ativação (que ficam acima das esteiras). Em seguida, o sistema deve ser executado: as esteiras vão levar os cristais pelas plataformas se esses forem iguais aos cristais de ativação, e consumidos em sequência. O jogador vence se o último cristal for consumido por um Automigo.

Portanto, todas as interações do jogador com os elementos de jogo estão estritamente relacionadas com o conteúdo de Autômatos Finitos Determinísticos, sendo equivalentes aos processos de construção de um autômato a partir de uma entrada e seu teste. As condições de vitória e derrota são apresentadas após a execução do sistema, que realiza o teste com a entrada, verificando se o autômato construído a aceita. A Figura 2 apresenta um Grafo de Fluxo de Controle (GFC) que ilustra essas condições.

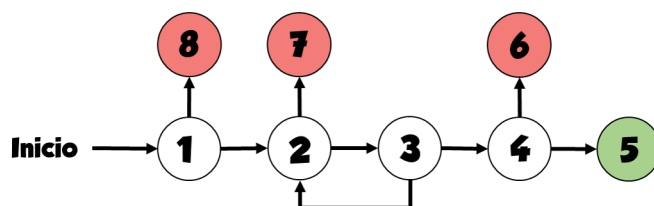


Figura 2. Condições de vitória e derrota do jogo.

Os nós 1, 2, 3 e 4 correspondem a momentos de verificação: 1) se o autômato possui estado inicial; 2) se o cristal mais à frente da fila é igual ao cristal de ativação; 3) se todas as entradas já foram consumidas; 4) se o estado final é um Automigo. Já os nós 5, 6, 7 e 8 são momentos que o usuário recebe um *feedback*: positivo (vitória) no nó 5 ou negativo (derrota) nos nós 6, 7 e 8. Esse *feedback*, além do resultado de vitória ou derrota, também contém uma quantidade de estrelas, que são condicionadas ao desempenho do jogador na fase (Figura 3). Assim, visando estabelecer um fluxo de aprendizagem, balanceando as recompensas do jogador e os desafios encontrados, estabeleceu-se que a dificuldade aumenta em cada fase, apresentando novos tipos de cristais e combinações.

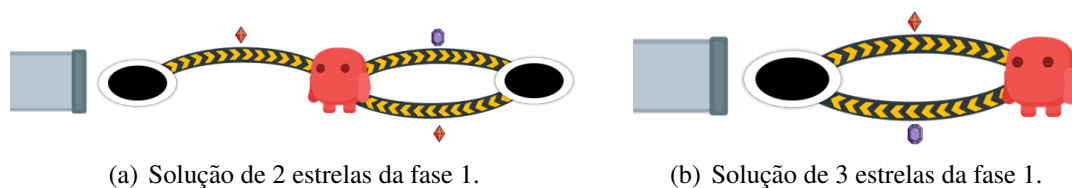


Figura 3. Possíveis soluções da primeira fase.

### 3.2.4. Teorias de aprendizagem

Quando se elabora um jogo educacional, diferente de um jogo com propósitos únicos de entretenimento, deve-se considerar tanto a diversão quanto os aspectos cognitivos [Pires et al. 2018]. Isso para que o conteúdo não fique tão implícito, a ponto do jogador ignorá-lo ou não praticá-lo, como também não sobrecarregá-lo e tornar o jogo maçante. Assim, faz-se necessário um alinhamento do jogo com as teorias de aprendizagem. Esse processo deve acompanhar todo o *game design*, desde a concepção até sua última avaliação, pois é o cerne que conecta a mecânica do jogo, o conteúdo trabalhado e o fluxo de aprendizagem.

Neste jogo, a teoria Construtivista, de Jean Piaget [Piaget 2003], é a que mais se encaixa. Essa teoria trata sobre a construção de conhecimentos do estudante a partir da interação com um objeto, cujo erro é enfatizado [Fino 2004, Sanchis and Mahfoud 2007, Pires et al. 2019]. Dessa forma, no contexto deste jogo, o jogador tem o objetivo de construir um autômato para avançar de fase, tendo a liberdade de elaborar suas próprias soluções. Ademais, o jogo o instiga a buscar por soluções mais viáveis, recompensando-o com mais estrelas se utilizar uma quantidade menor de estados. Assim, esse desafio fomenta o jogador a tentar e errar para atingir a perfeição numa fase, possibilitando-o exercitar ativamente o conteúdo proposto, no caso, a construção de AFD.

### 3.2.5. Pensamento Computacional

Também é importante que o jogo estimule o desenvolvimento do Pensamento Computacional (PC), uma habilidade tida como essencial para todos no século XXI e que diz respeito a reestruturar o pensamento para resolver problemas complexos de formas mais simples, cujo termo alavancou-se com a pesquisa de Wing [2006]. Existem várias divisões do PC, sendo a da BBC [2018] uma das mais utilizadas, que o estrutura em quatro pilares principais: decomposição – fragmentar um problema maior em subproblemas; reconhecimento de padrões – identificar recorrências/padrões nesses subproblemas; abstração – focar-se na parte mais relevante e; algoritmo – realizar um passo a passo lógico para solucionar o problema. Dessa forma, o *level design* do jogo foi estruturado a possibilitar que, enquanto jogue, o jogador esteja exercitando os quatro pilares do PC, como descrito na Tabela 2

### 3.2.6. Prototipação

Durante a elaboração de um jogo educacional, vários protótipos são construídos antes do produto final. Neste caso, três protótipos foram construídos: baixa, média e alta fidelidade. O protótipo de baixa fidelidade é geralmente um *paper prototype* e utilizado para uma validação inicial de mecânica, *design* e coesão com o conteúdo educacional. Por ser em versão de papel, possibilita ao desenvolvedor realizar modificações mais complexas e em tempo reduzido, quando comparado a uma versão já implementada, por exemplo. No âmbito deste trabalho, essas modificações resultaram em três versões desse protótipo – em que se utilizou o *Microsoft Power Point* para criação das artes.

O protótipo de média fidelidade também foi feito em papel, contudo, seu *design* foi melhorado através dos *softwares Adobe Photoshop e Power Point*, e o tipo de pa-

**Tabela 2. Pensamento Computacional no jogo.**

<b>Pilar do P.C.</b>	<b>Aplicação no jogo</b>
Decomposição	O jogador identifica os principais elementos de <i>gameplay</i> e os utiliza, de forma a: posicionar plataformas, conectar esteiras a plataformas, posicionar Automigos e trocar o cristal de ativação.
Reconhecimento de padrões	Após interações com o jogo, o jogador observa padrões, como: os cristais saem pelo tubo; o primeiro elemento dos cristais é comparado com o(s) cristal(is) da(s) esteira(s) conectada(s) e; a esteira é energizada se a comparação for verdadeira (cristais iguais).
Abstração	O jogador abstrai que: há várias combinações de cristais; é possível adicionar mais de uma esteira de saída em uma plataforma; quanto menos plataformas utilizar, maior a quantidade de estrelas recebidas e; para vencer, um Automigo deve consumir o último cristal.
Algoritmo	Por fim, após ter se habituado mais à <i>gameplay</i> , o jogador consegue construir um autômato com base na entrada (cristais) recebida e utilizar menos recursos (plataformas) no mesmo.

pel utilizado foi o couché, para maior qualidade. Essa versão estava mais fidedigna às especificações da documentação do jogo, contendo todas as telas, primeira fase, tutorial e *pop-up's* de erro. Já o protótipo de alta fidelidade é a versão mais robusta, em que se utiliza uma *game engine*. Fez-se uso da *Unity* e linguagem de programação *C#*, cujas falhas identificadas na versão anterior foram corrigidas. Essa versão ainda não é considerada um produto final, pois ainda carece da avaliação final com o público-alvo.

### **3.3. Avaliação**

Após a construção de cada protótipo, é necessário que o mesmo seja avaliado por usuários, para coletar *feedback*, possibilitando a identificação de falhas e obtenção de sugestões de melhorias. Como o protótipo de baixa fidelidade contou com várias versões, o mesmo foi validado tanto pelos autores quanto por estudantes da universidade. A primeira versão, por exemplo, possuía uma outra temática, que posteriormente foi alterada para espacial. Nas demais, as principais alterações deram-se no *storytelling* e na *User Interface (UI)*.

Quanto ao protótipo de média fidelidade, este foi avaliado por estudantes de cursos de computação durante a Semana Nacional de Ciência e Tecnologia (SNCT), cujos resultados são descritos na etapa seguinte. Em geral, o protótipo obteve uma boa aprovação em sua avaliação, feita através do teste MEEGA+ [Petri et al. 2019] via *Google Forms*, especialmente no *design*. Contudo, houve ressalvas, principalmente quanto à compreensão do conteúdo, que foram corrigidas na versão posterior. Já o protótipo de alta fidelidade, com o jogo funcional para dispositivos móveis, foi avaliado por estudantes de computação de outra universidade. Em suma, obteve uma boa avaliação, similar ao anterior, mas também apresentou ressalvas quanto ao entendimento do conteúdo, discutidas na próxima seção.

## **4. Resultados e discussões**

A ludificação do tema de Autômatos Finitos Determinísticos resultou num jogo educacional intitulado “Automigos”. De gênero aventura, ambientado em cenários espaciais e com visualização em duas dimensões (2D), o jogo promove o exercício do conteúdo de



AFD e o desenvolvimento do Pensamento Computacional, devendo o jogador construir um autômato que aceite a entrada para avançar de fase. Com quatro fases, cada uma correspondendo a um planeta dos Automigos, o jogo possui dois momentos (Figura 4): i) construção: o jogador constrói o autômato que aceite a entrada; ii) execução: o autômato é testado, verificando se a entrada é aceita.

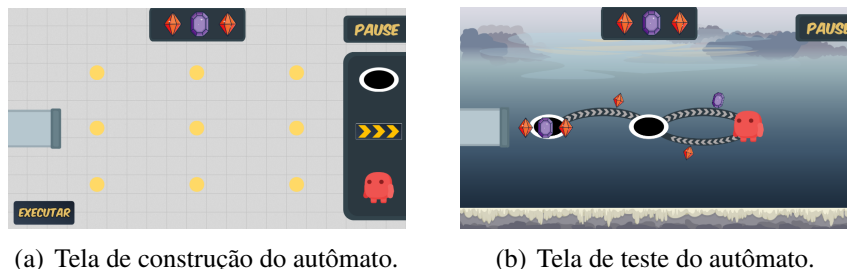


Figura 4. Construção e execução de um autômato na fase, respectivamente.

#### 4.1. Avaliações

Para verificar se o jogo atingia o seu propósito de facilitar o entendimento conteúdo de AFD e possibilitar que o jogador o pratique, os protótipos passaram por avaliações com diversos testadores. O protótipo de média fidelidade foi o que mais proveu resultados, cuja amostragem de testadores foi N=15, sendo dez do público masculino e cinco do feminino. Com exceção de um testador de 25 anos, a maior e menor idades encontradas no teste foram, respectivamente, 22 e 18 anos; ou seja, dentro da faixa etária pretendida do jogo. 11 dos 15 estudantes eram graduandos de cursos de computação (Licenciatura, Sistemas de Informação e Engenharia). O resultado dessas avaliações a partir do teste MEEGA+ é exibido na Figura 5.

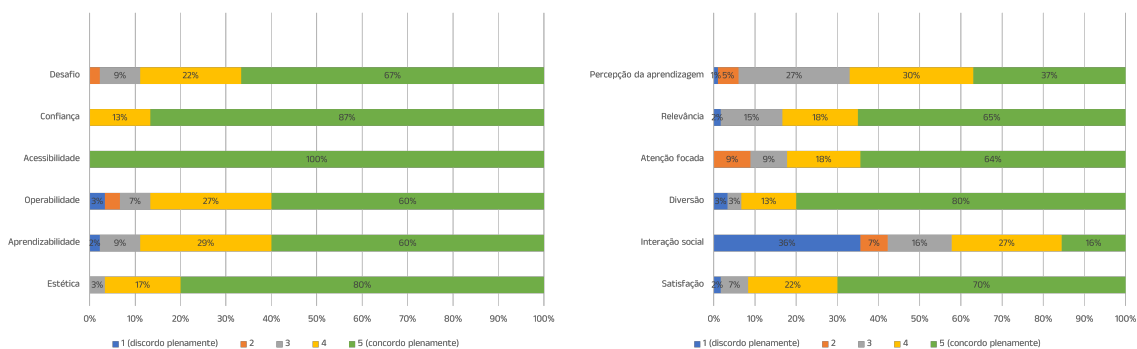


Figura 5. Resultados do MEEGA+ com o protótipo de média fidelidade.

Em geral, o jogo obteve bons resultados: das 12 métricas, 11 possuem suas melhores porcentagens no valor máximo de pontuação. A métrica de “interação social” obteve o pior resultado, pois o jogo não possui suporte para *multiplayer*. As melhores notas deram-se quanto à “Acessibilidade” e “Estética”, que dizem respeito ao *design* e organização visual do jogo (fontes, cores, tamanhos de botões, etc). No que se refere a “Percepção de aprendizagem”, observou-se uma porcentagem significativa que se manteve neutra, um indício de que o entendimento dos conteúdos estava um pouco complexo, que também se confirmou nas questões qualitativas, demandando a necessidade de um tutorial.

O protótipo de alta fidelidade, implementado em uma *game engine* e funcional para dispositivos móveis, não teve muitas modificações em relação à mecânica, quando comparado a sua versão anterior, que foram, em sua maioria, ajustes nas interfaces e adição de tutorial. Os testes com essa versão foram ligeiramente mais positivos, todavia, ainda não estava tão claro o entendimento das regras do jogo, mesmo com a inclusão do tutorial. Ademais, o jogo também apresentava algumas limitações: apesar de permitir que o jogador construa várias possibilidades de autômatos, há somente 12 campos para instanciar um estado e não é possível criar um auto-laço ou construir transições nas diagonais. Isso revelou que o jogo ainda necessitava de melhorias em seu próprio *game design*.

## 4.2. Lições aprendidas

Além das avaliações com o jogo, destacam-se algumas lições aprendidas pelos autores durante o processo da ludificação de um tema complexo de computação: *i) Questionamento sobre jogo ou gamificação* - Dúvida gerada após a finalização do trabalho, visto que o jogo possui mecânicas similares ao software *JFLAP*. Dessa forma, questiona-se se essa aplicação dos elementos de jogos foi uma gamificação do *JFLAP* ou um jogo, visto que foi elaborado do zero e que as mecânicas não são idênticas ao *software*. Tal questão ressalta a importância do estudo e compreensão de ambos os conceitos. *ii) O Game Designer* - O jogo é dependente de quem o faz. Ou seja, se o mesmo atingirá os objetivos de aprendizagem, depende de como foi elaborado o *learning design*, da criatividade e do conhecimento técnico das *hard skills* do autor, ou seja, o *game designer*. Assim, esse processo de *game design* educacional possibilita o desenvolvimento de competências multidisciplinares [Honda et al. 2020].

## 5. Considerações finais

Uma estratégia para facilitar a compreensão de um tema complexo de computação é sua ludificação. Neste trabalho, buscou-se ludificar Autômatos Finitos Determinísticos, que resultou em um jogo educacional que possibilita o exercício desse conteúdo e o desenvolvimento do Pensamento Computacional. Esse processo demandou uma série de etapas, e um conhecimento não somente do tema abordado, mas também dos aspectos cognitivos necessários para construir um jogo educacional.

Apesar do jogo ter sido bem avaliado, os resultados das avaliações apontaram inconsistências, inclusive, de *game design*. Esse é um fator que deve ser considerado, visto que, apesar do autor ser de um curso superior de computação, ter experiência com desenvolvimento de jogos educacionais e compreender o conteúdo de AFD, a abstração de seu funcionamento para implementá-la em uma *game engine* foi trabalhosa. Ademais, como o jogo ainda estava sendo desenvolvido, as avaliações tiveram como foco a usabilidade e percepção de aprendizagem, não tendo sido testado em disciplinas que abordam AFD para verificar se atingiu os objetivos de aprendizagem.

Dessa forma, a concepção de um jogo educacional a partir de um tema complexo de computação também consiste de uma atividade complexa e necessita de estudos minuciosos, além de recorrentes etapas de avaliação. Como trabalhos futuros, pretende-se: *i)* melhorar o jogo, a começar por uma reformulação no *game design*; *ii)* adicionar um sistema de captura de dados; *iii)* avaliar o jogo com estudantes de disciplinas relacionadas ao conteúdo de AFD e; *iv)* analisar os dados gerados e identificar as curvas de aprendizagem, possibilitando verificar se o mesmo atingiu seus objetivos de aprendizagem.

## Referências

- BBC (2018). Introduction to computational thinking. <https://www.bbc.co.uk/bitesize/guides/zp92mp3/revision/1>.
- Bennedsen, J. and Caspersen, M. E. (2019). Failure rates in introductory programming: 12 years later. *ACM inroads*, 10(2):30–36.
- Dognini, M. J. and Raabe, A. L. A. (2003). Eduling-software educacional para linguagens regulares. In *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)*, volume 1, pages 216–225.
- Fino, C. N. (2004). Convergência entre a teoria de vygotsky e o construtivismo/construcionismo. *acessado em*, 23(11):2016.
- Honda, F., Pires, F., Pessoa, M., and de Oliveira, E. H. T. (2020). Lições aprendidas em computação através da criação de um jogo educacional: entre automatos e design de aprendizagem. In *Anais do XXXI Simpósio Brasileiro de Informática na Educação*, pages 1753–1762. SBC.
- Honda, F., Pires, F., Pessoa, M., and Maia, J. (2022a). Cadê minha pizza? um jogo para exercitar matemática e pensamento computacional através de grafos. In *Anais Estendidos do XXI Simpósio Brasileiro de Jogos e Entretenimento Digital*, pages 876–885. SBC.
- Honda, F., Pires, F., Pessoa, M., and Melo, R. (2022b). Aplicando learning design na ludificação de percurso em grafos: uma jornada de aprendizagem. In *Anais do XXXIII Simpósio Brasileiro de Informática na Educação*, pages 609–620. SBC.
- Huizinga, J. (2014). *Homo ludens: A study of the play-element in culture*. Routledge.
- Jukemura, A. S., do Nascimento, H. A., and Uchôa, J. Q. (2005). Gam-um simulador para auxiliar o ensino de linguagens formais e de autômatos. In *XIII Workshop sobre Educação em Computação (WEI)*, pages 2432–2443.
- Macena, J., Pires, F., and Pessoa, M. (2020). Operação lovelace: uma abordagem lúdica para introdução de aprendizagem em algoritmos. *SBC-Proceedings of SBGames*.
- Macena, J., Pires, F., Pessoa, M., and Melo, R. (2022). Hello food: um jogo para praticar conceitos de algoritmos para iniciantes na computação. In *Anais Estendidos do XXI Simpósio Brasileiro de Jogos e Entretenimento Digital*, pages 1066–1075. SBC.
- McCulloch, W. S. and Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5:115–133.
- Melo, R., Pires, F., Lima, P., Pessoa, M., and de Oliveira, D. B. F. (2020). Metodologias para a criação de jogos educacionais: um mapeamento sistemático da literatura. In *Anais do XXXI Simpósio Brasileiro de Informática na Educação*, pages 572–581. SBC.
- Melo, R., Silva, D., and Pires, F. (2019). Stardust: um serious game para a aprendizagem implícita de grafos. In *Anais dos Workshops do Congresso Brasileiro de Informática na Educação*, volume 8, page 1237.
- Mioni, J. and Barbosa, C. (2022). Ferramentas para o aprendizado de linguagens formais e autômatos. In *Anais Estendidos do XXI Simpósio Brasileiro de Jogos e Entretenimento Digital*, pages 969–978. SBC.

- Neto, V. D. A. (2021). Uma análise sobre reprovação no curso de ciência da computação na ufrgs sob a ótica dos alunos.
- Petri, G., von Wangenheim, C. G., and Borgatto, A. F. (2019). Meega+: Um modelo para a avaliação de jogos educacionais para o ensino de computação. *Revista Brasileira de Informática na Educação*, 27(03):52–81.
- Piacentini, M. T. (2011). Jogo eletrônico, flow e cognição.
- Piaget, J. (2003). *The psychology of intelligence*. Routledge.
- Pires, F., Teixeira, K., Pessoa, M., and Lima, P. (2019). Desenvolvendo o pensamento computacional através da máquina de turing: o enigma do curupira. In *Anais do XXVII Workshop sobre Educação em Computação*, pages 523–532. SBC.
- Pires, F. G. d. S., Melo, R., Machado, J., Silva, M. S., Franzoia, F., and de Freitas, R. (2018). Ecologic: um jogo de estratégia para o desenvolvimento do pensamento computacional e da consciência ambiental. In *Anais dos Workshops do Congresso Brasileiro de Informática na Educação*, volume 7, page 629.
- Sanchis, I. d. P. and Mahfoud, M. (2007). Interação e construção: o sujeito e o conhecimento no construtivismo de piaget. *Ciências & Cognição*, 12.
- Santini, L., Junior, M. C., Felinto, A., and Aylon, L. (2022). Jogos no ensino de linguagens formais e autômatos: Um mapeamento sistemático. In *Anais Estendidos do XXI Simpósio Brasileiro de Jogos e Entretenimento Digital*, pages 886–895. SBC.
- Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 49(3):33–35.