

## INTEGRAÇÃO DE JOGOS ELETRÔNICOS NA EDUCAÇÃO E LAZER: DESENVOLVIMENTO E IMPACTO DE UM JOGO DE ESTRATÉGIA

Anderson Oliveira Ataíde<sup>1</sup>, José Rafael Franco<sup>2</sup>, Diego Augusto de Campos Moraes<sup>3</sup>

<sup>1</sup> Graduando em Análise e Desenvolvimento de Sistemas na FATEC Botucatu. e-mail: anderson.ataide@fatec.sp.gov.br

<sup>2</sup> Doutorando em Engenharia Agrícola, Faculdade de Ciências Agrônomicas - UNESP, BT. e-mail: jose.rafael@unesp.br

<sup>3</sup> Professor Doutor da Faculdade de Tecnologia de Botucatu. e-mail: Diego.Moraes11@fatec.sp.gov.br

### RESUMO

Este trabalho desenvolveu um jogo eletrônico de estratégia, que combina técnicas de jogos de cartas com elementos de raciocínio lógico, buscando ser uma ferramenta eficaz tanto para o lazer quanto para o aprendizado. O objetivo do jogo é estimular diversas habilidades cognitivas, como a capacidade de resolução de problemas, raciocínio lógico e a tomada de decisões estratégicas em contextos desafiadores. A metodologia utilizada foi uma pesquisa bibliográfica focando no desenvolvimento do jogo de estratégia. Os resultados obtidos indicam que, com o planejamento adequado e a adaptação às necessidades dos jogadores, a inclusão de jogos eletrônicos pode enriquecer significativamente a experiência de aprendizado, promovendo o desenvolvimento do raciocínio. A conclusão reforça o grande potencial dos jogos eletrônicos como uma abordagem inovadora e envolvente, tanto para fins educacionais quanto para entretenimento, proporcionando diversão e conhecimento.

Palavras-chave: Jogos eletrônicos, lazer, aprendizagem interativa.

### 1 INTRODUÇÃO

O uso da tecnologia para o aprendizado não é uma novidade; o ser humano sempre utilizou essa ferramenta para disseminação da informação e conhecimento (MONTES, 2024). Estudar sempre foi uma tarefa que demanda esforço, tempo e dedicação. O desenvolvimento de novas tecnologias, permite que surjam métodos inovadores que visam otimizar a forma como nos educamos. Embora possa parecer, à primeira vista, que jogos eletrônicos não são um meio eficaz de aprendizado, os jogos *online*, *offline*, sejam para dispositivos móveis, *desktop* ou qualquer outra plataforma, permitem adquirir conhecimento de forma eficiente e prazerosa, aumentando o interesse do jogador pelo assunto abordado a cada progresso no jogo.

Jogos eletrônicos focados no âmbito escolar, é uma ferramenta facilitadora de conhecimento com foco nos alunos e professores, mas até mesmo no lazer pessoal, o aprendizado pode ocorrer de várias maneiras, mesmo que a intenção principal não seja ensinar o jogador (ALVES, 2005). Brincar e interagir com a intenção de se divertir, possibilita que a criança aprenda regras e desenvolva aspectos cognitivos, sociais, afetivos e culturais. Portanto, este trabalho busca demonstrar como um jogo digital pode ser

explorado para lazer e aprendizado, utilizando elementos de estratégia por turnos de cartas.

## 2 MATERIAL E MÉTODOS

### 2.1 Metodologia Bibliográfica

Foi realizada uma pesquisa bibliográfica sobre o impacto dos jogos eletrônicos na educação. Conforme apontado por Moratori (2003), os jogos são atividades envolventes que atendem às necessidades das crianças, proporcionando um ambiente estimulante que desperta o interesse pelos desafios propostos pelas regras. Assim, as técnicas utilizadas para desenvolver uma jogabilidade estimulante são essenciais.

### 2.2 Ambiente de Desenvolvimento (IDE): Visual Studio Code

Como afirma o site oficial, *Visual Studio Code* (VSCode) é um editor de código versátil e amplamente utilizado, desenvolvido pela Microsoft. Ele se destaca pela sua facilidade de uso e adaptação às necessidades do desenvolvedor, permitindo a personalização por meio de extensões. Com suporte a diversas linguagens de programação, como Python, JavaScript, C++, entre outras, ele também oferece ferramentas de depuração, integração com sistemas de controle de versão como Git, e uma interface intuitiva.

A escolha do *Visual Studio Code* para o desenvolvimento do jogo, foi devido a suas diversas extensões, que facilitam na visualização do código, depuração de erros entre outros recursos tornando o processo de programação mais dinâmico, ágil e eficiente.

### 2.3 Draw.io para criação diagramas de caso de uso

Segundo o site oficial, *Draw.io* é uma ferramenta online gratuita utilizada para criar diversos tipos de diagramas, como fluxogramas, diagramas UML, diagramas de rede, ER (entidade-relacionamento) e muitos outros. Neste projeto, o Draw.io foi utilizado para modelar o sistema, criando diagramas de caso de uso que auxiliam na visualização da estrutura do jogo e na identificação de possíveis falhas.

O diagrama de caso de uso, visa criar um protótipo das possíveis ações do jogador mantendo o entendimento e contexto do projeto no início de sua construção e se necessário, criar mais diagramas atualizados (BACURAU; LEAL; RAMOS, 2022).

Essa abordagem proporcionou uma compreensão mais precisa do projeto e informações, prevenindo erros na criação do protótipo e na documentação das funcionalidades, o que facilita futuras alterações no jogo.

#### 2.4 Desenvolvimento do Jogo com Python

Python é uma linguagem amplamente usada em desenvolvimento web, análise de dados e automação (PYTHON,2024), foi criada em 1990 por Guido van Rossum, o nome “python”, foi inspirado no programa de comédia de televisão chamado “*Monty Python's Flying Circus*”. Ela descende da linguagem ABC, sendo de código aberto, fazendo bom uso do termo “*clean code*” por sua facilidade de compreensão e simplicidade. Ela é uma linguagem de alto nível para uso geral com possibilidade de reutilizar códigos de outras linguagens, ou seja, ao combinar C, C++, e Python, o programador pode criar soluções que são tanto rápidas quanto fáceis de desenvolver. (SILVA *et al.*, 2023).

O critério de escolha para utilização da linguagem Python, é sua facilidade no uso e aprendizagem, além de um desenvolvimento ágil e ótima compreensão das funções e orientações a objetos. No contexto do jogo de cartas, cada item do jogo possui um nome e um valor, o que torna a orientação a objetos mais prática para o desenvolvimento.

#### 2.5 Uso de bibliotecas: PySimpleGUI, Pygame

De acordo com o site oficial do *PySimpleGUI* (*PySimpleGUI*, 2024), é uma biblioteca do Python baseada na *TKinter*, uma das principais bibliotecas para criação de interface gráficas (GUIs). O *PySimpleGUI* foi desenvolvido com o objetivo de facilitar a criação de elementos de interface gráfica como, janelas, botões, caixas de texto entre outros componentes, de maneira simples, rápida e eficiente. A escolha do *PySimpleGUI* foi motivada pela necessidade de evitar a complexidade de bibliotecas tradicionais, como *Tkinter* ou *Qt*, e pelo desejo de desenvolver o front-end com poucas linhas de código.

Por outro lado, *Pygame* é uma biblioteca para criação de jogos e outras aplicações multimídia, baseada na *SDL* (*Simple DirectMedia Layer*). A biblioteca oferece funcionalidade para o gerenciamento de gráficos, som e entrada de dados de uma maneira relativamente fácil (*PYGAME*, 2024).

O jogo desenvolvido inclui elementos sonoros que tornam a experiência mais interativa, agregando características únicas ao projeto, para isso foi utilizado a biblioteca *Pygame* para o controle de mídia musical.

## 2.6 Integração com banco de dados: *SQLite*

Conforme o site oficial *SQLite* (2024), trata-se de um banco de dados SQL pequeno, rápido e confiável. É amplamente utilizado e seu código-fonte é público e pode ser utilizado por qualquer pessoa para qualquer propósito. A integração do *SQLite* neste projeto foi inspirada nos antigos jogos de fliperama, onde, ao ganhar ou perder, o jogador inseria seu apelido para que a máquina registrasse sua pontuação criando um ranking com os melhores jogadores. Esse sistema de pontuação permitia que outros jogadores tentassem superar o recorde.

No projeto foi implementado um sistema de pontuação semelhante. No final da partida, os pontos do ganhador são registrados na tabela de *ranking*. No entanto, em vez de o primeiro lugar ser ocupado por quem tem mais pontos, como em muitos jogos tradicionais, aqui o vencedor é aquele com a menor pontuação, de acordo com as regras do jogo.

O *SQLite*, permite que os dados sejam armazenados de forma segura onde o jogador possa consultar o *ranking* (tabela de pontos) os jogadores consultarem o *ranking* a qualquer momento sem o risco de perda algum dado, exceto se optarem por reiniciar o ranking para criar uma nova tabela de pontuações.

## 2.7 Métodos adotados para tornar o jogo dinâmico

Esse jogo utiliza elementos de cálculo de matemática básica e raciocínio lógico para estimular os jogadores a pensar estrategicamente em como vencer a partida. O jogo, foi projetado para dois jogadores, exige que cada um realize uma jogada por turno, com o objetivo de terminar com a menor pontuação. Cada carta possui um valor que, ao ser jogada, pode ser zerado, subtraído ou somado às outras cartas em campo determinado a pontuação final. Por se tratar de um jogo de cartas onde a estratégia é essencial para a vitória, foi necessário implementar mecânicas que permitissem equilibrar o jogo, favorecendo jogadores que estivessem em desvantagem. Para garantir esse equilíbrio, foi criada uma mecânica que permite "zerar" pontos: quando uma carta de valor igual é colocada na mesma coluna de outra carta, ambas passam a valer zero pontos. Dessa forma, um jogador com cartas de valores altos pode reduzir sua pontuação com uma jogada simples.

Figura 1 - Exemplo dos pontos zerados



Fonte: Dos autores

### 3 RESULTADOS E DISCUSSÕES

#### 3.1 Menu

Figura 2



Fonte: Dos autores

Figura 3



Fonte: Dos autores

O jogo apresenta um menu onde o jogador pode acessar diversas opções: Jogar, *Tutorial*, *Ranking*, Regras, Créditos e Sair. Esses elementos proporcionam ao jogo uma estrutura organizada, permitindo que o jogador consulte as regras e pratique no modo tutorial, o que melhora sua experiência e compreensão do jogo.

#### 3.2 Regras

A tela de regras desempenha um papel essencial no projeto, pois auxilia os jogadores a compreenderem melhor o jogo. Para facilitar esse entendimento, foram incluídos um

botão com narração em áudio, outro que direciona para um vídeo explicativo no *YouTube* e uma opção para relatar problemas. Assim, os jogadores podem informar erros por meio de um formulário, contribuindo para a correção de falhas e o recebimento de *feedbacks* para futuras atualizações.

### 3.3 Ranking

O *ranking* como é demonstrado na Figura 3, tem como função principal registrar e organizar os jogadores com base em suas pontuações, ordenando-os da menor para a maior pontuação, sendo que o primeiro colocado é o jogador com menor pontuação. Este sistema permite também a reinicialização da tabela a qualquer momento, caso os jogadores decidam iniciar uma nova rodada de pontuação.

### 3.4 Jogar

Antes de iniciar o jogo, é realizada uma verificação para garantir que os jogadores inseriram seus apelidos corretamente e não excederam o limite de caracteres no nome, permitindo o registro no *ranking*. Caso esteja tudo certo, os jogadores são direcionados para a tela de jogo, onde cada um recebe seis cartas viradas para baixo. No centro da tela, encontram-se dois montes de cartas: o monte de descarte, onde as cartas ficam com as faces viradas para cima, e o monte de compra, com as cartas viradas para baixo. Há também uma área vazia destinada a receber a carta comprada, que aguardará até a próxima jogada, completando o turno do jogador e passando a vez para o próximo.

Como pode ser visto na Figura 2, os montes de cartas são representados por vetores, sendo que no monte de compra uma função randômica escolhe uma carta de forma aleatória. No monte de descarte, as cartas são adicionadas ou retiradas, e em alguns casos, o monte pode ficar vazio.

Os nomes dos jogadores são exibidos ao lado de suas mesas, que são divididas por uma linha central. Ao lado dos nomes, aparece a pontuação de cada jogador, que é atualizada a cada jogada.

### 3.5 Restrições do jogo

Para evitar trapaças, durante o turno de cada jogador, as cartas do campo do oponente não podem ser alteradas. O jogador deve seguir as regras do turno, realizando a fase de compra de cartas e as substituições permitidas. Caso o jogador tente realizar qualquer

ação que não esteja de acordo com as regras, nada acontecerá e o jogo aguardará que ele complete sua jogada corretamente.

Se um jogador sair no meio da partida, seus pontos serão zerados e nenhum jogador será registrado no ranking. Uma nova partida poderá ser iniciada caso o jogador deseje jogar novamente.

### 3.6 Análise dos Resultados Obtidos

Durante o desenvolvimento e a implementação do jogo, foram realizados testes para verificar sua funcionalidade e a receptividade dos usuários. Os testes iniciais confirmaram que as principais funcionalidades do jogo, como o menu de navegação, a tela de regras e o sistema de *ranking*, estavam operando conforme o esperado. O *feedback* dos usuários ressaltou a eficácia dos recursos de suporte, como a narração em áudio e os vídeos explicativos, que facilitaram a compreensão e o engajamento com o jogo.

No entanto, os testes também identificaram a necessidade de ajustes nas mecânicas do jogo. Apesar da verificação de apelidos e limites de caracteres ter sido bem-sucedida na prevenção de erros de registro, alguns jogadores relataram dificuldades na interação com as cartas e na compreensão das regras durante o jogo. Essas questões foram abordadas com melhorias na interface gráfica e na clareza das instruções.

## 4 CONCLUSÕES

O projeto revelou a importância de uma interface intuitiva para melhorar a compreensão e a diversão do usuário. Ajustes contínuos baseados em feedback foram importantes para aprimoramentos na experiência dos jogadores e garantir um equilíbrio justo no jogo. A integração de elementos educacionais e de entretenimento no jogo destacou seu potencial para estimular o aprendizado e o pensamento estratégico de maneira envolvente e dinâmica.

## 5 REFERÊNCIAS

ALVES, Lynn *et al.* **Jogos eletrônicos e screenagens: possibilidades de desenvolvimento e aprendizagem.** In: SILVA, Eliane M.; MOITA, Filomena; SOUSA, Robson P. (Orgs.). **Jogos eletrônicos: construindo novas trilhas.** Campina Grande: EDUEP, 2007. Disponível em: [https://www.comunidadesvirtuais.pro.br/seminario-jogos/files/mod\\_seminary\\_submission/trabalho\\_188/trabalho.pdf](https://www.comunidadesvirtuais.pro.br/seminario-jogos/files/mod_seminary_submission/trabalho_188/trabalho.pdf). Acesso em: 03 jun. 2024.

**BACURAU, Rodrigo M.; LEAL, Brauliro G.; RAMOS, Ricardo A.** Uma abordagem para a construção de diagramas da UML concomitante à prototipação de interface. 2022. Disponível em: <http://www.univasf.edu.br/~brauliro.leal/pesquisa/Bacurau-5.pdf>. Acesso em: 06 jun. 2024.

**DRAW.IO. Diagrams.net.** Disponível em: <https://www.diagrams.net/>. Acesso em: 25 jun. 2024.

**MICROSOFT. Visual Studio Code Documentation.** Disponível em: <https://code.visualstudio.com/docs>. Acesso em: 15 jun. 2024.

**MONTES, P. V.** O impacto da tecnologia no processo de ensino-aprendizagem no ensino fundamental. *Epitaya E-books*, [S. l.], 2024. DOI: 10.47879/ed.ep.2024455p57. Disponível em: <https://portal.epitaya.com.br/index.php/ebooks/article/view/1075>. Acesso em: 03 jun. 2024.

**MORATORI, Patrick Barbosa.** Por que utilizar jogos educativos no processo de ensino aprendizagem. **UFRJ. Rio de Janeiro**, v. 4, 2003. Disponível em: [http://www.nce.ufrj.br/ensino/posgraduacao/strictosensu/ginape/publicacoes/trabalhos/t\\_2003/t\\_2003\\_patrick\\_barbosa\\_moratori.pdf](http://www.nce.ufrj.br/ensino/posgraduacao/strictosensu/ginape/publicacoes/trabalhos/t_2003/t_2003_patrick_barbosa_moratori.pdf). Acesso em: 24 maio 2024.

**PYTHON. Python.org.** Disponível em: <https://www.python.org/>. Acesso em: 12 jul. 2024.

**PYTHON SIMPLE GUI. PySimpleGUI: A simple, Pythonic GUI library.** Disponível em: <https://pysimplegui.readthedocs.io/>. Acesso em: 15 jul. 2024.

**PYGAME. Pygame: A set of Python modules designed for writing video games.** Disponível em: <https://www.pygame.org/>. Acesso em: 3 set. 2024.

**SILVA, Mateus Ataide et al.** Interfaces gráficas de usuários de maneira simples: o uso da biblioteca PySimpleGUI para criação de interfaces gráficas utilizando a linguagem **PYTHON SOFTWARE FOUNDATION.** Python Language Reference, versão 3.x. Disponível em: <https://www.python.org/>. Acesso em: 24 jul. 2024.

**SQLITE. SQLite: A software library that implements a self-contained, high-reliability, full-featured, SQL database engine.** Disponível em: <https://www.sqlite.org>. Acesso em: 18 ago. 2024.