



Universidade de Brasília – UnB
Faculdade de Ciências e Tecnologia em Engenharia – FCTE
Engenharia de Software

Math Hero: Um Jogo Educacional para o Ensino de Matemática

Autor: Matheus Pimentel Leal
Orientador: Dr. Edson Alves da Costa Júnior

Brasília, DF
2025



Matheus Pimentel Leal

Math Hero: Um Jogo Educacional para o Ensino de Matemática

Monografia submetida ao curso de graduação em Engenharia de Software da Universidade de Brasília, como requisito parcial para obtenção do Título de Bacharel em Engenharia de Software.

Universidade de Brasília – UnB

Faculdade de Ciências e Tecnologia em Engenharia – FCTE

Orientador: Dr. Edson Alves da Costa Júnior

Brasília, DF

2025

Matheus Pimentel Leal

Math Hero: Um Jogo Educacional para o Ensino de Matemática/ Matheus Pimentel Leal. – Brasília, DF, 2025-
62 p. : il. (algumas color.) ; 30 cm.

Orientador: Dr. Edson Alves da Costa Júnior

Trabalho de Conclusão de Curso – Universidade de Brasília – UnB
Faculdade de Ciências e Tecnologia em Engenharia – FCTE , 2025.

1. Engenharia de Games. 2. Engenharia de Software. I. Dr. Edson Alves da Costa Júnior. II. Universidade de Brasília. III. Faculdade UnB Gama. IV. Math Hero: Um Jogo Educacional para o Ensino de Matemática

CDU 02:141:005.6

Matheus Pimentel Leal

Math Hero: Um Jogo Educacional para o Ensino de Matemática

Monografia submetida ao curso de graduação em Engenharia de Software da Universidade de Brasília, como requisito parcial para obtenção do Título de Bacharel em Engenharia de Software.

Trabalho aprovado. Brasília, DF, 14 de Fevereiro de 2025:

Dr. Edson Alves da Costa Júnior
Orientador

Dr.^a Carla Silva Rocha Aguiar
Convidado 1

Dr. Matheus Bernardini de Souza
Convidado 2

Brasília, DF
2025

*Este trabalho é dedicado às crianças adultas que,
quando pequenas, se apaixonaram por games.*

Agradecimentos

Agradeço primeiramente a Deus, que me abençoou com a oportunidade de realizar este projeto que pode vir a causar grande impacto na vida de jovens e adolescentes, e também aos que estão apenas buscando novas maneiras de aprendizado. Por ter me abençoado com essa vida, por ter colocado tantas pessoas incríveis ao meu redor, por me dar saúde, por me proporcionar a oportunidade de servir por meio do meu trabalho e do meu conhecimento.

Agradeço imensamente a meus pais, Walderico de Fontes Leal e Alda Dias Pimentel Leal, pelo apoio incondicional, pelo amor, pela paciência, pelas horas dedicadas à minha educação, por terem me proporcionado uma infância cheia de amor, feliz e rica em amizades. Sem o apoio de vocês nada do que conquistei e nada do que sou seria possível. Obrigado Pai, por sempre acreditar no meu potencial, por me ensinar a ser um homem íntegro e ter bom coração. Obrigado mãe, por me ensinar a delicadeza, por ser minha maior incentivadora, por todo sacrifício feito, por todas as noites que a senhora passou em claro para cuidar de mim e por me mostrar que a resiliência e a paciência vencem qualquer obstáculo.

Agradeço ao meu irmão, Samuel Pimentel Leal, pelos anos de cuidado, companheirismo, carinho e amor. Obrigado por me ensinar que as diferenças entre as pessoas são o que as tornam únicas. Obrigado por me acompanhar e me proteger na minha inocência quando criança, por ter sido referência na busca por conhecimento, por me mostrar que sou mais capaz e corajoso do que imagino.

Agradeço à minha segunda mãe, Maria da Conceição da Silva, por ser inspiração de superação, de força, de coragem e de dedicação à Cristo. Por me mostrar que mesmo com grandes adversidades em nossas vidas ainda temos motivos para sorrir e acreditar. Por sempre cuidar de mim, por me amar mesmo não sendo teu filho de sangue e por me dar o enorme prazer de dizer à todos que encontro que tenho duas mães.

Agradeço à minha avó e madrinha, Maria Conceição Gomes Pimentel e minha tia Denise Dias Pimentel, por serem pilares de uma família amorosa e cheia de pessoas que procuram sempre fazer o bem. Obrigado por todos os momentos de amor compartilhados, por toda paciência comigo e por me terem me ensinado que o cuidado e o carinho para com os que amamos é incondicional.

Agradeço à minha noiva e futura esposa, Giulia Lobo Barros, com quem partilho a vida há 4 anos. Por ser minha parceira em todos os momentos, por me dar amor e carinho todos os dias, especialmente nos que não estou bem. Por me reintroduzir à minha

espiritualidade, por ser minha companheira na vida espiritual. Obrigado por me ajudar a carregar meus fardos, por me inspirar a ser melhor e por me mostrar que o dom do perdão é libertador.

Agradeço ao meu padrinho, José Carlos Leal, por cuidar de mim, me aconselhar, por ser presente na minha vida e por ser uma das maiores inspirações de paciência que tenho na vida.

Agradeço à toda a família da minha noiva por ter me recebido de braços abertos, pelos conselhos, pela ajuda, pelo carinho e por me proporcionarem alegria em todos os momentos que estamos juntos.

Agradeço aos irmãos que a vida me deu: Eduardo Souto Reis, Ricardo Cipriano Guimarães Gomes e Felipe Lademir Filippin. Vocês entraram na minha vida e por opção própria permaneceram e assim criamos o vínculo forte que temos hoje. Obrigado por serem meus melhores amigos de vida, pelos conselhos amorosos, pelos conselhos de vida, pela amizade, pelas risadas, por terem me abençoado com a amizade de vocês.

Obrigado à minha amiga, Daniela Ferreira de Oliveira, por ter me mostrado o inúmeros benefícios da psicologia, por não ter desistido de saber se eu estava bem até quando eu te afastei. Obrigado pelas inúmeras risadas, por ter sido luz em um momento de escuridão.

Agradeço ao meu parceiro de organização da *Game Jam FGA*, Liverson Paulo Furtado Severo, que dividiu comigo a organização deste evento que com tanto carinho idealizei. Nossas conversas e ideias me trazem inspiração para correr atrás de meus sonhos. Se não fosse pela ajuda dele, nem este trabalho e nem a *Game Jam FGA* tomariam forma.

Gostaria de agradecer também aos meus companheiros de FCTE: Ícaro Oliveira, Kalliu Brasil, Lude Ribeiro, Pedro Dias, Priscilla Rath e Lucas Régis. Cheguei na FCTE sem pretensão alguma de formar grandes laços de amizade e vocês foram uma surpresa incrível. Um agradecimento especial vai para o Ícaro, por me encher a paciência para trocar para o curso de Engenharia de Software, e, alguns anos depois, cá estou.

Agradeço ao professor Dr. Edson Alves da Costa Júnior, pelas incontáveis horas de parceria firmadas por meio da realização de trabalhos importantíssimos como este a *Game Jam FGA*. Obrigado pelos incontáveis conselhos, pela imensa paciência, pelo incentivo e pela grande inspiração que o Sr. se tornou em minha vida. Obrigado pela oportunidade de aprender contigo, seja nas aulas, seja por meio de projetos. Espero que eu possa um dia retribuir tudo o que o Sr. me proporcionou por meio do seu trabalho.

Por último, mas não menos importante, agradeço à toda *BsBeast's* por serem meus amigos e companheiros de jogatina, e também pelas risadas de todo dia no Discord.

*“ As grandes histórias nunca terminam realmente, Sr. Frodo.
Nem sempre quem as conta vê o fim delas,
e quando se vai embora, outra pessoa tem de continuar a contá-las.”
(O Senhor dos Anéis: O Retorno do Rei)*

Resumo

A dificuldade de alunos do ensino básico brasileiro na compreensão do currículo base de matemática não é novidade para os professores. Por isso diversos métodos pedagógicos vêm sendo utilizados a fim de melhorar a fixação dos conteúdos ensinados. Os *serious games* surgiram como proposta de metodologia lúdica para aumentar a fixação dos conteúdos por parte dos alunos, porém, o aspecto lúdico, que é responsável por proporcionar o divertimento dos jogadores, muitas vezes é deixado de lado. Este trabalho utiliza os conceitos da engenharia de *games* e da engenharia de software tradicional para a criação do *game* Math Hero, para auxiliar no ensino de técnicas de cálculo mental para alunos do ensino básico brasileiro. Tendo como base os conceitos citados, foi implementada uma versão *alpha* do jogo que contém uma mistura entre duas modalidades de modo de jogo *time attack* programadas para o desenvolvimento final, a modalidade competição e de treino, para que fosse possível a coleta, a análise e a comparação de dados de jogabilidade entre jogadores com e sem instrução prévia, no evento “Torneio de matemática mental” ocorrido durante a Semana de Extensão Universitária 2024 da Universidade de Brasília.

Palavras-chave: Engenharia de *Games*. Engenharia de Software. *Games*. Cálculo Mental. Matemática.

Abstract

The difficulty of Brazilian elementary school students have in understanding the basic math curriculum is not new to teachers. For this reason, various pedagogical methods are being used to improve the retention of the content taught. Serious games emerged as a play based methodology to increase students' retention of the math curriculum, but the fun aspect, which is responsible for providing players' enjoyment is often overlooked by the genre. This paper uses concepts from game software engineering and traditional software engineering to create the game *Math Hero*, to help teach mental math techniques to Brazillian elementary school students. Based on the aforementioned concepts, a alpha version of the game was implemented that contains a mixture of two game modes time attack programmed for the final development, the competition and training mode, so that it would be possible to collect, analyze and compare gameplay data between players with and without previous instruction, using the “Mental Math Tournament” event that took place during the University Extension Week 2024 at the University of Brasilia.

Key-words: Game Engineering. Software Engineering. Games. Mental Math. Mathematics.

Lista de ilustrações

Figura 1 – Imagem do <i>game</i> <i>Zelda: Tears of the kingdom</i>	20
Figura 2 – Processo de engenharia de requisitos	21
Figura 3 – <i>Game</i> Portal da Valve, em fase de <i>playtesting</i>	24
Figura 4 – Versão inicial do vilão do game <i>Math Hero</i>	40
Figura 5 – Versão inicial dos robôs de ajuda do game <i>Math Hero</i>	40
Figura 6 – Versão inicial do personagem principal do game <i>Math Hero</i>	41
Figura 7 – Tilemap provisório	41
Figura 8 – Estrutura de pastas <i>Math Hero</i>	44
Figura 9 – Tela de PC monocromática verde	47
Figura 10 – Paleta de cores <i>Gameflower green</i>	48
Figura 11 – Tela do modo <i>Time Attack</i>	48
Figura 12 – Evolução da mecânica utilizada no <i>time attack</i>	49
Figura 13 – <i>Scene</i> para realização de testes	49
Figura 14 – <i>Tilemap</i> de artes para construção visual do modo <i>time attack</i>	62

Lista de tabelas

Tabela 1 – Tabuada de multiplicações.	30
Tabela 2 – Classificação da busca de referências para o desenvolvimento do trabalho.	37
Tabela 3 – Expressões disponíveis para a operação de adição	45
Tabela 4 – Expressões $a - b$ disponíveis para a operação de subtração	46
Tabela 5 – Expressões $a \times b$ disponíveis para a operação de multiplicação	46
Tabela 6 – Expressões $a \div b$ disponíveis para a operação de divisão	47
Tabela 7 – Identificação das personas escolhidas para sessão de playtesting.	49
Tabela 8 – <i>Feedbacks</i> dados pelas pessoas escolhidas para a sessão de <i>playtesting</i>	50
Tabela 9 – Quadro de líderes de alunos de ensino básico treinados com as técnicas de matemática mental	52
Tabela 10 – Quadro de líderes de alunos de ensino superior sem treinamento prévio com as técnicas de matemática mental	53

Lista de abreviaturas e siglas

BNCC	Base Nacional Comum Curricular
EG	Engenharia de Games
ER	Engenharia de Requisitos
ES	Engenharia de Software
FCTE	Faculdade de Ciências e Tecnologias em Engenharia
FGA	Faculdade do Gama
GDD	<i>Game Design Document</i>
GSE	<i>Game Software Engineering</i>
MEC	Ministério da Educação
NPC	<i>Non Playable Character</i>
OCDE	Organização para a Cooperação e Desenvolvimento Econômico
SE	<i>Software Engineering</i>
UnB	Universidade de Brasília

Sumário

1	INTRODUÇÃO	15
1.1	Contexto	15
1.1.1	A matemática e o lúdico	15
1.1.2	O cálculo mental	16
1.2	Problema	16
1.3	Objetivos	17
1.4	Organização do trabalho	17
2	REFERENCIAL TEÓRICO	18
2.1	Jogos eletrônicos	18
2.1.1	Teoria de jogos	18
2.1.2	Game Design	19
2.2	Engenharia de software aplicada a videogames	19
2.2.1	Levantamento de Requisitos	20
2.2.2	Documento de <i>Game Design</i>	22
2.2.3	Implementação	23
2.2.4	<i>Playtesting</i>	23
2.3	Matemática mental	24
2.3.1	A multiplicação instantânea	25
2.3.2	A adição e subtração mental	26
2.3.2.1	Adição mental	26
2.3.2.2	Subtração mental	27
2.3.3	A multiplicação básica	29
2.3.4	A divisão mental	31
3	METODOLOGIA	34
3.1	Processo de desenvolvimento	34
3.1.1	Levantamento de requisitos	34
3.1.2	<i>Game Design</i>	34
3.1.3	Desenvolvimento	35
3.1.4	<i>Playtesting</i>	36
3.2	Pesquisa de fontes bibliográficas	36
4	RESULTADOS	38
4.1	Elicitação de requisitos	38
4.2	Game Design	39

4.2.1	Modo história	39
4.2.2	Modo <i>Time Attack</i>	42
4.3	Implementação	43
4.3.1	<i>Time Attack</i>	45
4.3.1.1	Escolha das referências	47
4.4	Playtesting	48
4.5	SEMEX 2024	50
4.6	Análise pós-implementação	53
5	CONSIDERAÇÕES FINAIS	55
5.1	Análise dos objetivos	56
5.2	Trabalhos futuros	57
	REFERÊNCIAS	58
	APÊNDICE A – IMPLEMENTAÇÃO DO <i>SINGLETON GAMEMANAGER</i>	61
	APÊNDICE B – <i>TILEMAP</i> DO MODO <i>TIME ATTACK</i>	62

1 Introdução

1.1 Contexto

1.1.1 A matemática e o lúdico

“A todo instante vivenciamos a matemática nas mais variadas situações – ela se faz presente na escola, em nossas casas, na economia, na política, na contagem do tempo e objetos, na compra e venda de mercadorias, no planejamento e controle do orçamento, ao brincar e jogar, etc. Enfim, somos cercados por números e formas e não há como negar a importância dessa área no nosso dia a dia. Portanto, é fundamental a contextualização, por parte dos professores, dos conteúdos matemáticos ensinados, estabelecendo um vínculo entre a origem teórica com as situações reais vivenciadas no cotidiano, pois apenas comentar onde e em que as operações são utilizadas não é o suficiente” (NASCIMENTO; LARA; TRUQUETE, 2023, p. 3).

Os problemas durante a aprendizagem da matemática entre crianças e adolescentes por vezes surgem por fatores como a falta de atenção dos alunos, o pré-condicionamento de alunos a acreditarem que a matemática é assunto praticamente impossível de ser compreendida ou até o despreparo didático por parte dos professores, como exposto na pesquisa feita por Emmel e Costa (2019). A necessidade da utilização de diferentes metodologias pedagógicas surgiu como tentativa de aumentar a assimilação dos conteúdos propostos pelo ensino básico. “Podemos dizer que, se bem utilizado, o jogo é uma metodologia para o ensino da matemática que pode auxiliar no desenvolvimento de habilidades de reflexão e discussão de ideias” (HERMANN; COQUEIRO; PACHECO, 2020, p.5).

Utilizar elementos lúdicos no meio pedagógico consiste basicamente em realizar atividades recreativas que despertem não somente a curiosidade, mas também a sede de aprendizado dos alunos participantes. Luckesi (2014) define a ludicidade como “um estado interno ao sujeito, ainda que as atividades, denominadas como lúdicas, sejam externas, observáveis e possam ser descritas por observadores, tais como os didatas, os historiadores, os sociólogos... A experiência lúdica (= ludicidade), que é uma experiência interna ao sujeito, só pode ser percebida e expressa pelo sujeito que a vivencia”.

Os jogos eletrônicos (*games*) são utilizados para atividades recreativas por pessoas de todas as faixas etárias. Em caráter específico à pedagogia, o gênero específico utilizado é o de *serious games*, que são games desenvolvidos com foco no caráter educacional, sem a função primária de oferecer entretenimento. Com esse foco no aspecto educacional, a maioria dos *serious games* possui baixa adesão dos estudantes, por deixarem de lado o aspecto lúdico.

1.1.2 O cálculo mental

“O cálculo mental consiste em diferentes estratégias que visam a obtenção de resultados por meio de métodos mais simples de resolução” (SILVA; GUALANDI; SANTOS, 2020, p. 62). Essa busca de métodos mais simples de cálculo consiste principalmente em dividir o problema inicial em problemas menores, mais fáceis de resolver.

“Compreendemos que uma boa Aritmética é aquela construída com cálculo mental. Com o cálculo mental o aluno desenvolve o senso numérico, a flexibilidade com os números, a criatividade, a autonomia e participa ativamente do processo de ensino, desenvolvendo uma base sólida em Aritmética. Além disso, o cálculo mental fortalece a relação do estudante com a Matemática, pois ele começa a ter sucesso em suas ações e escolhas na resolução de um problema”. (BERTICELLI; ZANCAN, 2023, p. 6)

1.2 Problema

Cerca de 73% dos alunos do ensino básico brasileiro registraram baixo desempenho segundo resultados do Pisa 2022 (OCDE, 2023). Os resultados divulgados pelo Ministério da Educação (MEC) mostram a baixa performance dos alunos do ensino básico brasileiro na disciplina de matemática. Esse resultado pode ser consequência, em parte, da grande complexidade dos conteúdos ministrados, aliada à escassez do ensino de técnicas para realização de cálculos mentais.

“Os documentos oficiais brasileiros trazem algumas orientações com relação ao desenvolvimento do cálculo mental no ensino fundamental. A Base Nacional Comum Curricular (BNCC), documento brasileiro que determina as aprendizagens essenciais, afirma que os estudantes devem desenvolver ao longo da Educação Básica, “diferentes estratégias para a obtenção dos resultados, sobretudo por estimativa e cálculo mental, além de algoritmos e uso de calculadoras” (BRASIL, 2017, p.224). No entanto, este documento não traz uma seção específica que explore o cálculo mental, ao contrário dos Parâmetros Curriculares Nacionais (PCN)” (COSTA, 2018).

O uso de técnicas de cálculo mental traz inúmeros benefícios aos alunos do ensino básico, não somente para utilização durante as avaliações, mas também para facilitar situações cotidianas, como o cálculo do valor de uma compra e a divisão de uma conta de restaurante, como apontado por Costa (2018, p. 46).

Entretanto, realizar o desenvolvimento de *games* com cunho educativo (do gênero *serious games*) pode se provar ser tarefa difícil tendo em vista que um dos desafios enfrentados pelo desenvolvedor é chegar à um conceito de jogo que seja divertido, cativante e que ao mesmo tempo não deixe de lado o propósito principal, que é fazer com que o público alvo aprenda o que precisa ser aprendido, como dito por Troyer e Janssens (2014,

p.1). A produção de um *game* nessas bases, ou seja, que possua um aspecto lúdico forte sem deixar de lado o viés educacional, possibilita o aumento da utilização de técnicas de matemática mental no currículo do ensino básico.

1.3 Objetivos

O objetivo deste projeto é realizar a conexão de elementos lúdicos ao ensino de técnicas de matemática mental para apoiar os pedagogos, além de alavancar as habilidades matemáticas de alunos que possuem dificuldade na realização de cálculos mentais ou na disciplina de matemática como um todo. Essa conexão será feita a partir do desenvolvimento do *serious game* Math Hero cujo público-alvo são jovens do ensino básico, com o objetivo de apresentar a eles uma forma leve e fácil de aprendizado de técnicas de cálculo mental.

Este trabalho também possui os seguintes objetivos específicos:

- aplicar padrões de *game design* atrativos para o público alvo;
- adicionar um modo de jogo competitivo que estimule o aprendizado e a competição saudável;
- coletar, no mínimo, os seguintes dados de jogabilidade do modo competitivo: tempo decorrido até a finalização, nome do jogador, configuração das questões e *seed* de jogo;
- desenvolver um modo de jogo que cativa o jogador por meio de elementos narrativos a aprender técnicas de matemática mental; e
- analisar os resultados obtidos a partir dos dados de jogabilidade dos jogadores.

1.4 Organização do trabalho

Este trabalho está organizado em seis capítulos. A [Introdução](#), que realiza a contextualização do leitor ao projeto; o [Referencial Teórico](#), que descreve e demonstra os principais conceitos atrelados ao trabalho; a [Metodologia](#), onde é apresentado o plano metodológico seguido por este trabalho e que caracteriza o projeto do game *Math Hero*; os [Resultados](#), onde estão a análise e a disposição dos resultados alcançados com a finalização e aplicação do projeto; e, por fim, [Considerações Finais](#), onde é disposta a análise final do trabalho assim como considerações sobre o seu desenvolvimento e evolução.

2 Referencial Teórico

Este capítulo apresenta conceitos da engenharia de software, de *videogames* e de técnicas de matemática mental e interliga estes assuntos a fim de criar uma base teórica sólida para a realização deste trabalho, além de trazer ao leitor os conceitos necessários para o entendimento do mesmo.

2.1 Jogos eletrônicos

Para definir o que é um jogo eletrônico, ou um *videogame*, como são mais popularmente conhecidos, primeiramente é necessário definir o que é um jogo.

Rogers (2010) define um jogo como uma atividade que requer no mínimo um jogador, tem regras e possui uma condição de vitória. Alguns exemplos simples de jogos que se encaixam nesta definição são: rebater uma bola na parede, arremessar um objeto o mais longe possível e memorizar a ordem de uma série de imagens.

Um *videogame* é um modo de interação entre um jogador, uma máquina com um *display* visual eletrônico, e possivelmente outros jogadores, que é mediada por meio de um contexto ficcional significativo para o jogador, e sustentada pelo apego emocional entre o jogador e os resultados de suas ações nesse contexto ficcional (ARJORANTA, 2019, p.109 - 110, tradução nossa).

O desenvolvimento de *videogames* é um processo multidisciplinar, composto por pessoas das mais diversas áreas, como artistas, programadores e até advogados.

A constante evolução da indústria de jogos faz com que conceitos importantes para a padronização de processos e produtos finais (os jogos) surjam. Uma análise mais profunda sobre os conceitos de teoria de jogos e de *game design*, que são pilares para o entendimento deste trabalho, é de grande utilidade.

2.1.1 Teoria de jogos

A teoria de jogos é uma importante área da matemática que introduz ferramentas que tornam possível a análise de situações compostas por agentes que tomam decisões baseadas nas ações de outros agentes.

“A teoria de jogos pode também ser definida como o estudo sobre como o resultado final de uma situação competitiva é guiado pelas interações entre as pessoas envolvidas no jogo (também podem ser chamados de ‘jogadores’ ou ‘agentes’), baseado nos objetivos e preferências desses

jogadores e na estratégia adotada por cada um deles” (HUANG, 2010, p.2, tradução nossa).

Por estratégia entende-se apenas as diferentes maneiras, e por consequência, diferentes cadeias de ações que um agente pode adotar para alcançar um objetivo.

2.1.2 Game Design

Depois da popularização dos jogos eletrônicos como meio de entretenimento, a especialização da indústria em diferentes áreas deu-se de forma natural, visto que, com mais jogos sendo feitos, surge a necessidade de padronizar e evoluir todo o processo que permeia a produção de um *game*. Segundo Rogers (2010), o profissional de *game design* é responsável por criar as ideias e regras que compreendem o jogo, de modo a tornar possíveis diferentes estratégias que possam ser utilizadas pelos jogadores em diferentes partes do jogo. “Quando pessoas jogam, elas têm uma experiência. O designer se importa justamente com essa experiência. Sem a experiência, o jogo não tem valor.” (SCHELL, 2008, p.10, tradução nossa).

A área de *game design* se trata exclusivamente da experiência do jogador dentro do *game*, criando desde ambientes imersivos e interessantes para o jogador durante o ciclo completo do jogo.

2.2 Engenharia de software aplicada a *videogames*

Segundo Sommerville (2015), a Engenharia de Software (ES) é uma disciplina da engenharia que está relacionada com todo o processo produtivo de um software, desde os estágios iniciais de especificação do sistema até a manutenção depois do início de seu uso. E, por meio da utilização de métodos, ferramentas e teorias, procura trazer manutibilidade, confiabilidade, segurança, eficiência e aceitabilidade ao produto de software.

“Videogames são produtos de software avançados, e, ao mesmo tempo, são produções complexas de criatividade e arte. Essa união de disciplinas, a necessidade de o produto ser divertido e o crescimento implacável da indústria de videogames, criaram diferenças significativas entre a engenharia de software tradicional e a engenharia de software para games (GSE)” (CHUECA et al., 2024, p. 1, tradução nossa).¹

Entretanto, vale lembrar que em alguns aspectos a engenharia de software tradicional e a engenharia de software para games (ou somente, engenharia de games) possuem

¹ “Video games are advanced software products and, at the same time, they are complex works of creativity and art [3]. This merge of disciplines, the need for the product to be fun, and the unrelenting growth of the video game industry have created significant differences between traditional Software Engineering (SE) and Game Software Engineering (GSE)”

similaridades significativas, de modo que a aplicação de conceitos, métodos e ferramentas da engenharia de software tradicional sejam inevitáveis.

Figura 1 – Imagem do *game* Zelda: Tears of the kingdom



Autor: Nintendo. Fonte: Google Imagens

Um dos aspectos dos quais a engenharia de games utiliza, apesar das diferenças, é a adoção de métodos de desenvolvimento de software. As metodologias se dividem em dois tipos, as metodologias tradicionais e as metodologias ágeis. No âmbito do desenvolvimento de games, as metodologias ágeis são preferencialmente adotadas, tendo em vista sua alta iteratividade e baixo foco na documentação, o que oferece facilidade ao time de desenvolvimento de um jogo para realizar atividades como o *playtesting*, que consiste em realizar testes de versões específicas do jogo com jogadores selecionados para obter *feedbacks* rápidos e corrigir erros e comportamentos indesejados antes do lançamento oficial.

No contexto de engenharia de games, é comum separar atividades com responsabilidades específicas para padronizar o processo e seguir o método de desenvolvimento escolhido pela equipe. Algumas das atividades comuns na engenharia de games são o levantamento de requisitos, o *game design*, a implementação e o *playtesting*.

No game Zelda: Breath of The Wild, os conceitos da GSE possibilitou a identificação de melhorias no game design durante sessões de *playtesting*. As melhorias foram tão significativas que foram levadas como aprendizado para a criação do game que dá sequência à este, o *Zelda: Tears of The Kingdom* (Figura 1).

2.2.1 Levantamento de Requisitos

“As pessoas e organizações têm desejos e necessidades de coisas a serem construídas ou de serem evoluídas. Chamamos essas necessidades de requisitos” (IREB, 2022, p. 9 - *syllabus* do nível fundamental, CPRE). O objetivo da Engenharia de Requisitos

(ER) é realizar o gerenciamento e o levantamento dos requisitos do sistema para que o usuário tenha suas necessidades e desejos realizados. A Figura 2 descreve as etapas de uma iteração do processo de engenharia de requisitos.

Figura 2 – Processo de engenharia de requisitos



Autor: PNS (com adaptações). Fonte: Google Imagens

É comum realizar a utilização de técnicas que auxiliam no levantamento, priorização e gerenciamento dos requisitos de um projeto. Para a elicitação (levantamento) dos requisitos, podem ser utilizadas técnicas de coleta e técnicas geradora de ideias e desenhos.

“As técnicas mais comuns utilizadas em elicitação de requisitos são as de introspecção, entrevista e análise de protocolos. A técnica de introspecção se baseia em imaginar que tipo de sistema eu iria querer se eu estivesse executando esta tarefa, utilizando este equipamento, entre outros. (...) As entrevistas podem ser divididas em entrevistas com questionários, entrevistas livres e grupos focais e de desenvolvimento de aplicações. (...) Já a técnica de análise de protocolos pede a um indivíduo que se envolva em uma tarefa, explicando ao mesmo tempo o seu processo de raciocínio” (MORAES, 2008).

A escolha correta das técnicas de elicitação é fundamental para o sucesso de um projeto, visto que, caso os requisitos levantados estejam errados, o projeto não irá cumprir

com as necessidades e vontades do usuário final. Para realizar a escolha correta, o tipo do sistema, o método de desenvolvimento de software, as pessoas envolvidas e a configuração organizacional devem ser levados em consideração.

Uma técnica de extrema importância é a priorização de requisitos, uma vez que em projetos de software devemos sempre levar em consideração o prazo dado, o tamanho da equipe e a capacitação da mesma para a execução de tarefas de desenvolvimento. Uma das técnicas mais difundidas entre os profissionais de engenharia de requisitos é a priorização MoSCoW, que consiste em separar os requisitos em diferentes categorias de priorização.

“O nome *MoSCoW* é derivado das primeiras letras dos critérios de priorização a seguir:

Must have (Mo), *Should have* (S), *Could have* (Co), *Want to have but will not have this time round* (W).

Para a maioria dos profissionais, “W” na verdade significa “Won’t have”. Técnicas de ranqueamento e técnicas de votação cumulativa são utilizadas para identificar as necessidades que precisam ser resolvidas, em oposição às necessidades das quais os *stakeholders* gostariam que fossem resolvidas. Os casos de uso definidos para o sistema podem ser explicitamente associados com a necessidade resolvida por eles. Isso permite uma avaliação mais objetiva do benefício proporcionado por cada caso de uso e garante que cada caso de uso realmente esteja ajudando a sanar uma necessidade do *stakeholder*.” (BITTNER; SPENCE, 2003, p.74).

2.2.2 Documento de *Game Design*

Segundo Pedersen (2003), o profissional de *game design* é o visionário. É como o autor de um livro, que determinou o escopo e a descrição do produto com detalhes suficientes para que outros (o restante do time de desenvolvimento do game) possam entender e desenvolver o produto. Por esse motivo, uma das principais atribuições do *game designer* dentro do time de desenvolvimento é ser o criador e escritor das ideias do game em forma de um documento de *game design*.

“O documento de *game design* descreve tudo o que estará no jogo. É um documento muito importante que será consultado por toda a equipe durante a produção do jogo” (ROGERS, 2010, p. 72, tradução nossa)² e, por isso, ele reflete detalhadamente a ideia do jogo, a arte, a trilha sonora, as mecânicas e até os diálogos entre os personagens existentes no game. Levando em consideração a necessidade do time de desenvolvimento, o tempo limite para a produção do game e o orçamento disponível, essa documentação pode tomar formas diferentes, desde um documento formal e padronizado até uma simples apresentação de *slides*.

² “A *GDD* outlines everything that will be in the game. It’s a very important document that the entire team will refer to during the production of your game.”

2.2.3 Implementação

“O processo de desenvolvimento é algo que todo estúdio tem que seguir se ele quer criar um bom *game*. Esse processo é iterativo, isso significa que algumas funcionalidades podem parecer boas quando escritas em papéis mas quando criadas não funcionam ou não são tão boas como esperado” (CARMONA, 2016, p. 10 tradução nossa).³

Portanto, o processo de desenvolvimento é de extrema importância para a criação de bons *games*, pois ela reflete diretamente o compromisso do time de desenvolvimento com a criação, o refino e a evolução do game.

Dentro do processo de desenvolvimento, a implementação é a fase central, onde é feita a construção das ideias propostas nas fases de levantamento de requisitos e de *game design*. É por meio dela que o *game* toma forma para que a iteratividade do processo de desenvolvimento seja respeitada e também para que o maior número de erros e comportamentos indesejados sejam identificados e corrigidos antes de alcançar o jogador.

2.2.4 Playtesting

“*Playtesting* é simplesmente observar pessoas jogarem uma parte do jogo, por vezes com um questionário ou entrevista ao final. E depois usar o que você observa e escuta, para poder realizar mudanças no design do game” (Mark Brown - GMTK, 2023, tradução nossa)⁴. Segundo Swift (2007), desenvolvedor da *Valve* no game *Portal* (Figura 3), provavelmente, a coisa mais importante aprendida por ele foi o processo de realização de sessões de *playtesting*.

A fase de *playtesting* precisa ser tratada como prioridade no desenvolvimento de um *game*, pois nela são colhidos os principais *feedbacks* sobre o comportamento atual e sobre o *game design*. A aplicação de um questionário ou de uma entrevista direcionada ao final de uma sessão de *playtesting* faz com que o time de desenvolvimento consiga colher respostas específicas para aplicar mudanças e melhorias no *game*.

Mas ainda assim, muitas respostas podem ser colhidas apenas observando o jogador durante a sessão de *playtesting*. Dificuldades encontradas pelo jogador durante o *game*, falhas na visualização de objetos cruciais à progressão, ou até a frustração passada por ele durante o game podem ser importantes para a realização de ajustes no design do jogo.

³ “The development process is something that every studio has to follow if they want to create a good game. This process is iterative, this means that some features could sound good when written in papers but then when created are not working properly or they are not as good as expected.”

⁴ “Playtesting is simply watching people play through a chunk of the game, sometimes with a questionnaire, or interview at the end. And then using what you see and hear to drive changes in the game’s design.”

Figura 3 – *Game Portal da Valve, em fase de playtesting*

Autor: Valve. Fonte: Google Imagens

2.3 Matemática mental

“É consenso, entre os professores da educação básica, as dificuldades enfrentadas ao ministrar aulas de matemática, não só pela complexidade dos conteúdos, mas também pela difícil assimilação de conceitos pelos alunos” (NASCIMENTO; LARA; TRUQUETE, 2023). Segundo dados coletados pelo Instituto Nacional de Estudos e Pesquisas (INEP), por meio do Sistema de Avaliação do Ensino Básico (SAEB) 2021, 55,3% dos alunos do ensino fundamental brasileiro pertencem aos 4 níveis inferiores de proficiência de matemática, do total de 8 níveis nos quais os alunos são avaliados.

Como apontado por Benjamin e Shermer (2006, tradução nossa), “muitas vezes, a matemática é ensinada como um conjunto rígido de regras, deixando pouco espaço para pensamento criativo”⁵. Com isso, muitos alunos podem acabar sendo desmotivados e até pensar que habilidades matemáticas são inatas. Mas, como toda habilidade, ela pode ser aperfeiçoada e treinada por meio do aprendizado de técnicas facilitadoras.

“A base da Matemática é a aritmética. Alicerçado na aritmética temos praticamente todos os outros conhecimentos matemáticos. Para compreender a subtração, é necessário primeiro, compreender bem a adição. Para compreender a multiplicação é necessário ter conhecimentos de adição e subtração. Da mesma forma, para compreender a divisão, é necessário conhecimentos de adição, subtração e multiplicação. Assim, para que o aluno tenha um bom entendimento e uma boa relação com a Matemática, é imprescindível que ele saiba aritmética” (BERTICELLI; ZANCAN, 2023).

Diversas técnicas são ensinadas por Benjamin e Shermer (2006) no livro *Secrets of mental Math: The Mathemagician’s guide to lightning fast calculation and amazin math*

⁵ “Too often, math is taught as a set of rigid rules, leaving little room for creative thinking.”

tricks, com o objetivo de despertar o interesse e a curiosidade do leitor, para que, com as técnicas aprendidas, este possa não só impressionar amigos, familiares e colegas, mas também melhorar suas habilidades aritméticas. As principais técnicas apresentadas no livro que serão abordadas neste trabalho são a multiplicação instantânea, adição e subtração mental, multiplicação básica e divisão mental. Para melhor abstrair os conteúdos dos métodos ensinados, é necessário saber que o princípio fundamental da matemática mental é ter uma visão de resolução de problemas, transformando um problema grande em alguns outros problemas mais fáceis de serem solucionados.

2.3.1 A multiplicação instantânea

Uma das técnicas mais interessantes é a multiplicação de qualquer número de dois dígitos por 11. Assim que ela se torna conhecida, fica muito fácil realizar esse tipo de operação, dá a quem aprende a sensação de grande proficiência em matemática e causa grande admiração por parte de quem vê essa técnica sendo utilizada.

Utilizando as técnicas apresentadas por [Benjamin e Shermer \(2006\)](#), para se multiplicar qualquer número de dois dígitos por 11, basta somar os dois dígitos, caso o resultado seja menor que 10, e colocar o resultando entre os dígitos originais, da seguinte forma:

$$32 \times 11$$

$$3 + 2 = \mathbf{5}$$

$$32 \times 11 = \mathbf{352}$$

Agora, suponha que a soma dos dois dígitos seja maior que 10: como proceder? Nesse caso, devemos somar 1 ao primeiro dígito do número original, e colocar normalmente o segundo dígito entre os dígitos originais. Vejamos o exemplo abaixo:

$$85 \times 11$$

$$8 + 5 = \mathbf{13}$$

$$85 \times 11 = \mathbf{935}$$

Um outro exemplo seria: 57×11 . Como $5 + 7 = 12$, o resultado é 627. Similarmente, uma conta muito comum é 99×11 . Utilizando a técnica ensinada, podemos fazer: $9 + 9 = 18$, obtendo:

$$\begin{array}{r} 1 \\ 989 \\ \hline 1089 \end{array}$$

Também é possível utilizar este método em números de três dígitos. Por exemplo, para o número 314, a resposta ainda começa com 3 e termina com 4. E, como $3 + 1 = \underline{4}$ e $1 + 4 = \underline{5}$, o resultado é igual a 3454.

2.3.2 A adição e subtração mental

Utilizando a técnica de adição e subtração da esquerda para a direita, é possível realizar cálculos matemáticos mais rapidamente para uma grande variedade de números encontrados no dia a dia.

“Existem diversas boas razões pelas quais é melhor realizar cálculos matemáticos mentais pensando da esquerda para a direita. Afinal, nós fazemos a leitura de números da esquerda para a direita, pronunciamos seus nomes da esquerda para a direita, então, é simplesmente mais natural pensar (e realizar cálculos) da esquerda para a direita” ⁶(BENJAMIN; SHERMER, 2006, p. 12, tradução nossa).

2.3.2.1 Adição mental

Os problemas de adição mais fáceis são os que não requerem que um número seja carregado para a outra casa (o “vai um”). Por isso, o pensamento da esquerda para direita será demonstrado a partir deste problema simples até problemas mais complexos. Tomemos como exemplo a seguinte expressão: $47 + \underline{32}$. Para dividir o problema em problemas menores, podemos escrever a mesma adição na forma $47 + \underline{30} + \underline{2}$ e, por consequência, pensar da seguinte forma:

$$\begin{array}{r} 47 \\ + 30 \\ \hline 77 \end{array}$$

Por fim, basta solucionar o problema mais simples (adicionar 2 ao resultado parcial) para obter o resultado final:

$$77 + 2 = 79$$

O diagrama a seguir exemplifica o processo de pensamento:

$$\begin{array}{ccccc} 47 + 32 & = & 77 + 2 & = & 79 \\ & \text{(primeiro adicionar 30)} & \text{(depois adicionar 2)} & & \end{array}$$

Todo este processo pode ser utilizado para solucionar expressões matemáticas onde há a necessidade de que um número seja carregado para a próxima casa. Na prática, pode-

⁶ “There are many good reasons why it is better to work from left to right. After all, you read numbers from left to right, you pronounce numbers from left to right, and so it’s just more natural to think about (and calculate) numbers from left to right.”

mos tomar como exemplo a expressão $67 + 28$. Utilizando um diagrama para representar o processo de pensamento, podemos solucionar a expressão da seguinte forma:

$$\begin{array}{rclclcl} 84 + 57 & = & 134 + 7 & = & 141 \\ & \text{(primeiro adicionar 50)} & & \text{(depois adicionar 7)} & \end{array}$$

Para realizar a adição de números de três dígitos, seguimos o mesmo processo de pensamento, realizando a divisão do problema original em problemas menores. Assim, podemos computar o resultado da adição $538 + 327$. Usando o diagrama de processo mental, podemos executar o cálculo da seguinte maneira:

- primeiro adicionar 300: $538 + 327 = 838 + 27$
- depois adicionar 20: $838 + 27 = 858 + 7$
- por fim, adicionar 7: $858 + 7 = 865$

Todos os problemas de adição mental podem ser resolvidos desta maneira. Basta continuar dividindo o problema inicial em problemas menores, e solucionar cada um destes. Quando realizamos essa divisão em pequenos problemas, primeiramente é recomendado que os dados sejam ditos em voz alta e depois, com a prática, realizar a solução deles mentalmente. O fato de registrarmos mentalmente os resultados das operações, nos dispensa a necessidade de memorizar o “vai um” para soma-lo ao resultado final.

2.3.2.2 Subtração mental

Para realizar subtrações, os mesmos princípios e técnicas serão utilizados para facilitar o desenvolvimento e a resolução de expressões. Um problema simples seria:

$$\begin{array}{r} 85 \\ -25(20 + 5) \\ \hline \end{array}$$

Utilizando a mesma técnica e processo de pensamento demonstrados na adição, podemos separar o problema em duas partes mais simples de serem resolvidas: $85 - 20$, e depois, $65 - 5$, o que nos daria **60** como o resultado final. Esta abordagem se torna possível ao aplicar o pensamento da esquerda para a direita, se atentando primeiramente aos algarismos mais significativos.

Porém, expressões matemáticas de subtração são mais difíceis quando o problema exige que números “emprestem” um à casa adjacente. Tomando como exemplo a expressão $86 - 29$, podemos tomar a resolução de duas diferentes formas:

- Primeiro subtrair 20, depois 9:

$$86 - 29 = 66 - 9 = 57$$

- Primeiro subtrair 30, depois adicionar 1:

$$86 - 29 = 56 + 1 = 57$$

Uma regra de uso geral para a escolha entre as duas forma diferentes é analisar se o problema faz necessário o “empréstimo”, arredondar o segundo número para o próximo múltiplo de 10 e depois adicionar a diferença.

Agora, analisando subtrações de três dígitos, a resolução da expressão $958 - 417$ é simples por não requerer nenhum “empréstimo” para ser solucionada:

- Primeiro subtrair 400: $958 - 417 = 558 - 17$
- Depois subtrair 10: $558 - 17 = 548 - 7$
- Por fim, subtrair 7: $548 - 7 = 541$

Agora, olhando para uma expressão que exige um “empréstimo”, podemos solucioná-la da seguinte forma para diminuir a quantidade de passos e facilitar a resolução:

$$\begin{array}{r} 747 \\ -598(600 - 2) \\ \hline \end{array}$$

- Primeiro subtrair 600: $747 - 598 = 147 + 2$
- Por fim, adicionar a diferença: $147 + 2 = 149$

Estes problemas exemplificam o caso onde o segundo número da expressão é próximo de um múltiplo de 100. Em expressões onde não é esse o caso, a resolução se torna mais complicada. Por exemplo, considere a expressão $725 - 468$. Podemos encontrar a diferença entre o segundo número e o próximo múltiplo de 100 (no caso, **500**), utilizando complementos:

$$\begin{array}{r} 57 \\ +43 \\ \hline 100 \end{array} \quad \begin{array}{r} 21 \\ +79 \\ \hline 100 \end{array} \quad \begin{array}{r} 49 \\ +51 \\ \hline 100 \end{array} \quad \begin{array}{r} 68 \\ +32 \\ \hline 100 \end{array}$$

Pode-se observar que, para cada um dos pares de números adicionados, o resultado é 100. Isso se dá porque quando adicionamos os primeiros números (os da esquerda), o resultado é 9, e para os últimos (os da direita) o resultado é 10. Com isso, dizemos que 43 é o complemento de 57, 32 é o complemento de 68 e 51 é complemento de 49. É importante notar que, como tudo que foi apresentado no livro *Secrets of Mental Math* de Benjamin e Shermer (2006), toda resolução passa pelo princípio básico de solucionar equações da esquerda para a direita.

Retornando ao problema $725 - 468$, podemos visualizá-lo agora utilizando complementos para chegar à uma solução mais fácil:

$$\begin{array}{r} 725 \\ -468(500 - 32) \\ \hline \end{array}$$

Assim, o passo a passo de resolução mental seria:

- Primeiro subtrair 500: $725 - 500 = 225$
- Depois adicionar 30: $225 + 30 = 255$
- Por fim, adicionar 2: $255 + 2 = 257$

2.3.3 A multiplicação básica

O pré-requisito de qualquer problema envolvendo expressões matemáticas de multiplicação é saber a tabuada de multiplicação de 1 a 10, como apresentada na Tabela 1. Esse conhecimento se torna fundamental para o entendimento e para que seja possível realizar mentalmente o cálculo de expressões de multiplicação. É importante também saber aplicar a propriedade distributiva, o que facilita a visualização da resolução dos problemas de multiplicação básica.

Aliando o conhecimento do processo de resolução da esquerda para a direita com as técnicas de adição e subtração mentais e o conhecimento da tabuada do 1 ao 10, torna-se possível realizar o cálculo de multiplicações de forma mental e, assim, facilitar e agilizar a resolução de problemas matemáticos no dia a dia.

Utilizando o pensamento da esquerda para a direita, e também o princípio de simplificação do problema para auxiliar na resolução, é possível resolver a seguinte expressão matemática de multiplicação: 42×7 .

O problema pode ser simplificado e resolvido em 3 passos:

- Primeiro, multiplicar 40 por 7: $42 \times 7 = 280 + (2 \times 7)$

Tabela 1 – Tabuada de multiplicações.

×	1	2	3	4	5	6	7	8	9	10
1	1	2	3	4	5	6	7	8	9	10
2	2	4	6	8	10	12	14	16	18	20
3	3	6	9	12	15	18	21	24	27	30
4	4	8	12	16	20	24	28	32	36	40
5	5	10	15	20	25	30	35	40	45	50
6	6	12	18	24	30	36	42	48	54	60
7	7	14	21	28	35	42	49	56	63	70
8	8	16	24	32	40	48	56	64	72	80
9	9	18	27	36	45	54	63	72	81	90
10	10	20	30	40	50	60	70	80	90	100

Fonte: o Autor

- Depois, multiplicar 2 por 7: $280 + (2 \times 7) = 280 + 14$
- Finalmente, somar os resultados: $280 + 14 = 294$

Este problema não requer que um número seja “carregado” (vai um) para a próxima casa e, por isso, é um problema de simples assimilação. Porém, existem expressões matemáticas que possuem esta necessidade e portanto são mais complexos como, por exemplo,

$$\begin{array}{r} 67 \\ \times 8 \\ \hline \end{array}$$

Mesmo que pareça complexo, ao seguir o processo de mental da esquerda para a direita e o princípio da simplificação, o problema se torna:

- Multiplicar 60 por 8: $67 \times 8 = 480 + (7 \times 8)$
- Multiplicar 7 por 8: $480 + (7 \times 8) = 480 + 56$
- Por último, realizar a soma dos resultados: $480 + 56 = 536$

É possível também utilizar a técnica de arredondamento para facilitar a resolução de uma dada expressão matemática. Vejamos o seguinte exemplo: 78×9 . O processo mental continua o mesmo, porém, ao arredondarmos o número para cima, para facilitar a conta, obtemos:

$$78 \times 9 = (80 - 2) \times 9$$

$$\begin{aligned}(80 - 2) \times 9 &= 80 \times 9 - 2 \times 9 \\ 80 \times 9 - 2 \times 9 &= 720 - 18 = 702\end{aligned}$$

Ou, de forma análoga, podemos calcular o mesmo problema com outro arredondamento:

$$\begin{aligned}78 \times 9 &= 78 \times (10 - 1) \\ 78 \times (10 - 1) &= 78 \times 10 - 78 \times 1 \\ 78 \times 10 - 78 \times 1 &= 780 - 78 = 702\end{aligned}$$

O arredondamento com subtração funciona particularmente bem quando o número em questão está apenas a dois dígitos de distância do próximo múltiplo de 10, caso contrário, o método com adição funciona melhor. Isso se dá ao fato de que a subtração fica mais complexa que a adição nestes casos, e já que a rapidez é essencial ao cálculo mental, isso deve ser levado em consideração na resolução.

No caso de multiplicações de um número de três dígitos por um número de um dígito, o passo a passo não é diferente, apenas aumentando a quantidade de passos para certos problemas. Vejamos o exemplo:

$$\begin{array}{rcl} & 326(300 + 20 + 6) & \\ \times & 7 & \\ \hline 2100 & = 300 \times 7 & \\ + 140 & = 20 \times 7 & \\ \hline 2240 & & \\ + 42 & = 6 \times 7 & \\ \hline 2282 & = 326 \times 7 & \end{array}$$

No exemplo acima, a solução é composta de 3 passos. Primeiro resolvermos a expressão 300×7 , depois 20×7 e, por fim, 6×7 . Com isso, basta somar os resultados para obter a resposta final para o problema.

2.3.4 A divisão mental

“A divisão Mental é uma habilidade particularmente conveniente de se ter, tanto no trabalho quanto na vida cotidiana”⁷...

A habilidade de dividir na sua mente pode te poupar da inconveniência de ter de pegar uma calculadora toda vez que você precisa de calcular algo.”⁸ (BENJAMIN; SHERMER, 2006, p. 80, tradução nossa)

⁷ “Mental division is a particularly handy skill to have, both in business and in daily life”

⁸ “The ability to divide in your head can save you the inconvenience of having to pull out a calculator every time you need to compute something”

Na técnica de divisão mental, o pensamento da esquerda para a direita funciona naturalmente, porque é assim que o ensino básico escolhe ensinar a resolução de problemas de divisão. Por isso o foco será na demonstração de técnicas que facilitam realizar contas de divisão mentalmente, e não no processo mental, visto que este já está enraizado em nosso conhecimento.

Benjamin e Shermer (2006) explicam que o primeiro passo para a divisão mental é descobrir quantos dígitos estarão presentes na solução. Para isso, considere o seguinte problema:

$$179 \div 7$$

Nesse caso, estamos procurando um número Q tal que $7 \times Q = 179$. Sabemos que, $7 \times 10 = 70$ e $7 \times 100 = 700$, então, o número Q deve ser um número entre 10 e 100. Com isso, podemos concluir que a solução é um número de 2 dígitos. Em seguida, determinamos o próximo múltiplo de 10 que, multiplicado por 7, resulta em um número menor que 179. Para isso, temos que: $7 \times 20 = 140$, e $7 \times 30 = 210$. Logo, podemos dizer que a solução Q está entre 20 e 30. Depois, como esta parte da solução não mudará, podemos subtrair $179 - 140 = 39$.

Por fim, sabendo que $7 \times 5 = 35$ e que 35 tem uma diferença de 4 para 39, o resultado final é 25 com resto 4, ou ainda:

$$Q = 25 + \frac{4}{7}$$

Todo processo de resolução de expressões de divisão pode ser simplificado, independentemente do tamanho. Portanto, podemos observar que, um problema $179 \div 7$ pode ser simplificado para um problema muito menor: $39 \div 7$. Este processo de simplificação pode e deve ser utilizado para facilitar o cálculo de toda expressão de divisão, incluindo problemas onde o dividendo é um número de três dígitos. Tomando como exemplo a expressão

$$947 \div 4$$

Podemos solucioná-la da mesma forma, simplificando o problema em problemas menores e pensando da esquerda para a direita. Neste caso, sabemos que $4 \times 100 = 400$, e que $4 \times 1000 = 4000$, logo o resultado possui 3 dígitos. Em seguida, deveremos encontrar o maior múltiplo de 100 que, pode ser subtraído de 947, que neste caso é $4 \times 200 = 800$. Assim, sabemos que a resposta com certeza começa com **2**. E, subtraindo $947 - 800 = 147$, nos deparamos com outro problema, desta vez mais simples. Como $4 \times 30 = 120$, sabemos que o próximo dígito da solução é **3**, e podemos simplificar ainda mais o problema

subtraindo $147 - 120 = 27$. Por fim, como $4 \times 6 = 24$, sabemos que o último dígito da solução é o **6** e, podemos encontrar o resto da divisão realizando a subtração $27 - 24 = 3$. Logo, a solução fica sendo **236** ou, de outra forma,

$$947 \div 4 = 236 + \frac{3}{4}$$

3 Metodologia

Considerando que um dos objetivos deste trabalho é unir elementos lúdicos ao ensino de matemática para jovens e crianças, este capítulo apresenta o plano metodológico em relação aos seguintes tópicos: métodos de desenvolvimento de software, levantamento de requisitos, ambiente de desenvolvimento, fontes de pesquisas bibliográficas e a localização e escolha dos *assets* do *game*.

3.1 Processo de desenvolvimento

O processo de desenvolvimento é parte vital para que um software em fase de produção possa atingir altos níveis de qualidade e, conseqüentemente, sucesso. A iteratividade deste processo é importante durante o desenvolvimento de um *game*, visto que ideias que pareçam geniais, quando propostas, podem ser abandonadas, ou até terem seu desenvolvimento prorrogado, por não suprirem expectativas, ou até não funcionarem como o esperado dentro do contexto do *game*.

Nesta seção serão abordadas e exemplificadas todas as fases do desenvolvimento do *Math Hero*.

3.1.1 Levantamento de requisitos

Durante essa etapa, as técnicas escolhidas para a elicitação dos requisitos foram a introspecção e a entrevista. A técnica de introspecção trouxe liberdade ao processo criativo, para que este trabalho pudesse seguir sem adotar padrões de projetos de *games* educativos já existentes, visto que estes geralmente possuem baixa adesão por parte de seu público alvo, por deixarem o caráter lúdico de lado em favor do caráter educacional. Já a técnica de entrevista foi utilizada principalmente para coletar *feedbacks* dos jogadores e dos observadores, para que fossem feitos ajustes tanto na parte de jogabilidade (*gameplay*) quanto nos elementos que proporcionam a sensação de imersão ao jogador durante todo fluxo do *game*.

3.1.2 *Game Design*

Para criar um *game*, normalmente inicia-se definindo um objetivo a ser alcançado pelo jogador. E, para tornar sua jornada até este objetivo divertida e imersiva, o *Game Design* define o escopo e as mecânicas que o jogador terá acesso, bem como os elementos

gráficos e sonoros para auxiliar, guiar e imergir o jogador na superação de obstáculos até alcançar o objetivo final.

Primeiramente, para conhecer melhor o que era desejado, foi necessário definir o que era não era desejado. E, a partir do distanciamento de conceitos de games educativos padrão, foi definido que o game *Math Hero* teria como principal objetivo não deixar o aspecto lúdico de lado, em favor do aspecto educativo. Seria necessário então progredir nesta fase com foco em balancear o educativo e o lúdico dentro do *game*.

O gênero escolhido foi o *platformer adventure puzzle* para que, durante o modo história, o jogador possa aprender técnicas matemáticas e também utilizar suas habilidades para progredir. Além disso, quando este estiver jogando o modo *time attack*, ele poderá testar suas habilidades e competir contra colegas em desafios matemáticos com temporizadores.

3.1.3 Desenvolvimento

A fase de desenvolvimento de um videogame, como todo software, é onde ocorre a união dos requisitos, regras e funcionalidades levantadas nas fases anteriores. A escolha de ferramentas e linguagens de programação adequadas altera drasticamente a experiência de desenvolvimento, o fluxo de trabalho e também a qualidade do produto final.

A primeira etapa feita na fase de desenvolvimento foi justamente a escolha de ferramentas de desenvolvimento que favorecessem este processo, tendo como ponto de partida para as escolhas a experiência prévia do autor com desenvolvimento de jogos. Assim, foi definido que as ferramentas utilizadas no desenvolvimento deste projeto seriam o git 2.34.1, GitHub, Godot Game Engine 4.3, Visual Studio Code 1.96.4 e Aseprite v1.3-beta.

A escolha feita pela Godot Game Engine foi baseada nos requisitos do *game*, por ser uma *game engine* que é gratuita, de uso livre e de código aberto, além de trazer facilidade quanto ao aprendizado e utilização dos recursos, por possuir uma extensa e completa documentação de seus recursos nativos, e por ter uma comunidade grande de desenvolvedores, o que faz com que seja possível buscar por ajuda em fóruns e comunidades virtuais de desenvolvedores.

Logo em seguida, para seguir as boas práticas da engenharia de *games*, foi necessário realizar a escolha dos métodos de desenvolvimento de software a serem seguidos, para melhor rastreabilidade e controle das tarefas realizadas durante a produção. Assim, foi definida a utilização do SCRUM solo, exemplificado por Pagotto et al. (2016), além de elementos do XP (Extreme programming) (WELLS, 2009).

O planejamento do desenvolvimento do game foi feito por meio da plataforma web

de colaboração e controle de versionamento GitHub, por meio da funcionalidade GitHub Projects. As etapas de planejamento adotadas foram, em sequência: a definição do escopo e do tema do *game* por meio de reuniões com o professor orientador deste projeto, a definição da duração das *sprints* do projeto, a frequência das reuniões de acompanhamento, e também a rotina de acompanhamento do projeto pelo professor orientador.

A duração de uma *sprint* foi definida para duas semanas, tendo em vista o tempo necessário para a criação ou seleção de *assets*, desde *sprites* 2D até arquivos de áudio para trilha sonora, e para o desenvolvimento da mecânica planejada. A periodicidade das reuniões de acompanhamento foi semanal, para que o professor orientador pudesse ter maior visão e controle da fase atual de desenvolvimento. E, por fim, a estrutura de acompanhamento do andamento das tarefas de desenvolvimento foi feita por meio de um quadro automático que envia atualizações para o e-mail do professor orientador sobre novas tarefas, atualizações em tarefas existentes e finalização de tarefas planejadas.

3.1.4 *Playtesting*

A fase de *playtesting* é de extrema importância para o desenvolvimento de um *game*, pois ela possibilita a visualização e adaptação de mecânicas desenvolvidas, bem como a exclusão e ideação de novas mecânicas que se encaixem melhor no contexto definido para o *game*. Mesmo em fase de *playtesting*, a equipe de desenvolvimento de um jogo pode aprender muito e realizar correções e mudanças que, sem o *feedback* de pessoas externas ao desenvolvimento, não seriam feitas.

Durante a prototipação, os *feedbacks* coletados de pessoas pertencentes ao público alvo serviram para obter uma visão geral de como a ideia do *game* se comportaria perante os usuários finais. As mudanças sugeridas impactaram não somente o *game design* mas as mecânicas propostas também, para que o conceito do game se tornasse mais atrativo para jovens e crianças do ensino básico.

3.2 Pesquisa de fontes bibliográficas

A pesquisa de fontes bibliográficas é parte crucial de um trabalho de caráter científico. Por isso, a seleção de fontes que auxiliam na escrita, desenvolvimento e produção de resultados torna-se uma tarefa extensa e importante.

Para este trabalho, por conta do tema principal, as fontes bibliográficas escolhidas variam desde documentações à vídeos informativos disponibilizados no [YouTube \(2023\)](#). Essa grande variedade de fontes informativas foi necessária devido ao caráter multidisciplinar do desenvolvimento de jogos. Mas, também, foi necessário uma fundamentação teórica sólida para que fosse possível demonstrar como práticas propostas pela engenharia

de *software* são aplicadas ao desenvolvimento de *games*, além de demonstrar o impacto da adição de elementos lúdicos ao ensino da matemática.

Os termos pesquisados foram “Game Development, Game Engineering, Game Design, Cálculo Mental, Ludicidade e Ensino, Requisitos de Software, SCRUM, Desenvolvimento Ágil, Jogos e Matemática, Aprendizado matemática, Extreme Programming e IREB”. Dos resultados encontrados, foram selecionados 7 artigos de relevância para este trabalho. A Tabela 2 representa os motores de busca utilizados, os termos de busca, a quantidade de resultados, a quantidade de artigos avaliados e a quantidade escolhida. Os critérios para a seleção foram a relação com o tema deste trabalho e a relevância da referência.

Tabela 2 – Classificação da busca de referências para o desenvolvimento do trabalho.

Motor de busca	Termos	Resultados	Avaliados	Escolhidos
Periódicos CAPES	<i>Game Engineering</i>	144.265	15	5
Periódicos CAPES	<i>Game Development</i>	73.866	13	2
Periódicos CAPES	<i>Game Design</i>	72.484	4	1
Periódicos CAPES	<i>Cálculo Mental</i>	818	4	2
Periódicos CAPES	<i>Ludicidade, ensino</i>	736	10	2
Periódicos CAPES	<i>Requisitos de Software</i>	946	3	2
Periódicos CAPES	<i>SCRUM</i>	4.456	10	1
Periódicos CAPES	<i>Desenvolvimento ágil</i>	312	4	3
Periódicos CAPES	<i>jogos; matemática</i>	1.572	10	3
Periódicos CAPES	<i>Aprendizado matemática</i>	1.142	4	1
YouTube	<i>Game Engineering</i>	Não disponível	10	0
Youtube	<i>Game Development</i>	Não disponível	3	1
YouTube	<i>Game Design</i>	Não disponível	25	3
Youtube	<i>Cálculo Mental</i>	Não disponível	5	0
Google	<i>Extreme Programming org</i>	79.600.000	15	1
Google	<i>IREB</i>	582.000	5	1

Fonte: o Autor

4 Resultados

O objetivo deste capítulo é realizar a exposição dos resultados obtidos por meio da realização deste trabalho, com ênfase nos objetivos alcançados. Os tópicos apresentados estão ordenados com base na execução das tarefas do projeto, sendo elas a [Elicitação de requisitos](#), o [Game Design](#), o [Implementação](#) e o [Playtesting](#). As últimas seções, denominadas [SEMEX 2024](#) e [Análise pós-implementação](#), apresentam os resultados obtidos por meio da utilização do *game* Math Hero em trabalhos realizados durante a semana de extensão da UnB no ano de 2024.

4.1 Elicitação de requisitos

Na etapa de elicitação dos requisitos do projeto foi utilizada a técnica MoSCoW de priorização para definição da priorização de tarefas de desenvolvimento. Com base nos parâmetros da técnica, foram definidos, inicialmente, os requisitos descritos no Quadro 1.

Quadro 1 – Requisitos iniciais do *Game* Math Hero.

Requisito	Priorização
O <i>game</i> Math Hero deve possuir código aberto	<i>Must Have</i>
O <i>game</i> Math Hero deve facilitar a adição de plugins e personalizações	<i>Should Have</i>
O <i>game</i> Math Hero deve utilizar artes 2D	<i>Must Have</i>
O <i>game</i> Math Hero deve possuir boa didática	<i>Must Have</i>
O <i>game</i> Math Hero deve utilizar linguagem acessível ao público alvo	<i>Must Have</i>
O <i>game</i> Math Hero deve possuir um Modo história para aprendizado de técnicas de matemática mental	<i>Must Have</i>
O <i>game</i> Math Hero deve possuir um modo <i>Time Attack</i> para competições entre jogadores	<i>Must Have</i>
O <i>game</i> Math Hero deve ser multiplataforma (mobile e desktop)	<i>Should Have</i>
O <i>game</i> Math Hero não deve ter monetização associada	<i>Must Have</i>
O <i>game</i> Math Hero deve possuir um modo tutorial para aprendizado das mecânicas do <i>game</i>	<i>Should Have</i>
O <i>game</i> Math Hero deve ensinar técnicas de matemática mental	<i>Must Have</i>

Fonte: o Autor

Como o processo de desenvolvimento de um jogo guiado pela engenharia de games é iterativo, ocorreram, ao longo do desenvolvimento do jogo, ocorreram, ao longo do desenvolvimento do jogo, mudanças nos requisitos descritos no Quadro 1. Foram removidos os requisitos referentes ao desenvolvimento do modo história do jogo, sendo eles:

- o *game* Math Hero deve possuir boa didática;
- o *game* Math Hero deve possuir um modo história para aprendizado de técnicas de matemática mental;
- o *game* Math Hero deve possuir um modo tutorial para aprendizado das mecânicas do game; e
- o *game* Math Hero deve ensinar técnicas de matemática mental.

Desde a versão inicial foi feita a transposição dos requisitos para tarefas que pudessem ser priorizadas e organizadas em um quadro de planejamento, além da criação do *roadmap* do projeto, Estas tarefas e estes quadros foram atualizados e adaptados ao longo de todo o desenvolvimento, para refletir as decisões e adaptações necessárias.

4.2 Game Design

“Frequentemente quando estivermos fazendo um game, começamos inserindo um objetivo ao qual o jogador deve alcançar. Mas, como levar o jogador do início do game até este objetivo, e mais importante, como fazer com que o jogador se divirta na jornada, é onde a parte complexa, mas divertida, começa” (BRACKEYS, 2007, min 1:10, tradução nossa).¹

Na fase de *game design* foi definido que o game Math Hero possuiria 2 modos jogáveis: um modo história, onde o jogador percorre um mundo fictício repleto de atividades relacionadas à matemática, onde por meio do aprendizado e aplicação de técnicas de matemática mental, o jogador poderia melhorar suas habilidades matemáticas e se divertir ao mesmo tempo; e um modo *time attack*, que seria principalmente para competições, contendo métricas de números de erros e tempo gasto, mas também tem o intuito de estimular que os jogadores utilizem habilidades de matemática mental para solucionar questões matemáticas relacionadas às quatro operações básicas: adição, subtração, divisão e multiplicação.

4.2.1 Modo história

Para o modo história do game foi definido como objetivo final do jogador: participar de um duelo matemático com um cientista maluco (Figura 4) para que ele pare com a loucura das expressões “proibidas”. Para isso, o jogador deveria percorrer o laboratório do Professor R para retirar dali todos os seguidores do cientista maluco e recuperar o

¹ “Often when making a game, you start by setting a goal that the player has to achieve. But how to get the player from the beginning of the game and towards that goal, and most importantly, how to get the player to enjoy this journey, is where the tricky, and also fun part, comes into play”

laboratório do professor, que está cheio de expressões matemáticas proibidas escritas em seus quadros (como divisões por zero, por exemplo).

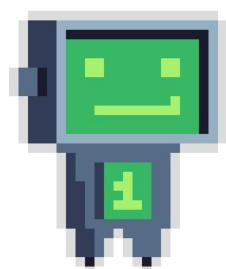
Figura 4 – Versão inicial do vilão do game *Math Hero*



Fonte: o Autor

Por meio da conversa com os robôs (Figura 5) de ajuda do Professor R, o jogador aprenderia técnicas de matemática mental úteis no dia a dia, e também valiosas para superar os obstáculos (como portas travadas, enigmas nos quadros, itens trancados com senha, e mais) deixados pelos seguidores do cientista maluco nas salas do laboratório do Professor R.

Figura 5 – Versão inicial dos robôs de ajuda do game *Math Hero*



Fonte: o Autor

Depois de definir o objetivo final e a jornada do jogador, foi preciso definir o estilo de arte, para que o game fosse atrativo para crianças e adolescente do ensino básico brasileiro. O estilo de arte escolhido foi o *pixel art*, que traz facilidade de assimilação por parte dos jogadores por proporcionar simplicidade visual ao usuário final, além de

facilitar a criação e animação dos *assets* do game. O personagem principal (Figura 6) do game é um(a) jovem aprendiz do Professor R.

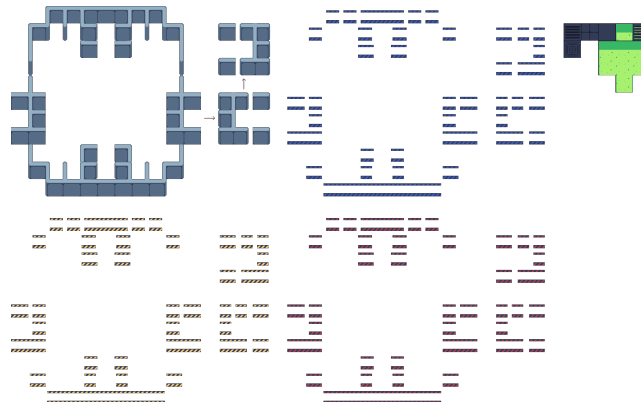
Figura 6 – Versão inicial do personagem principal do game *Math Hero*



Fonte: o Autor

Na fase de level design do protótipo do game *Math Hero*, a primeira etapa foi a criação e busca de *assets* provisórios para desenvolver rapidamente as mecânicas propostas pela fase de elicitação de requisitos e de game design. O primeiro passo desta etapa foi a criação do *sprite* do personagem principal do jogo utilizando o software Aseprite². Em seguida, foi necessário realizar a busca de um *tilemap* provisório que atendesse às necessidades levantadas nos requisitos. Buscando por palavras chave como “2D, *Science*, *Lab* e *Robots*”, foi possível encontrar um *tilemap* que supriu as necessidades momentâneas do game (Figura 7).

Figura 7 – Tilemap provisório



Fonte: o Autor

A implementação do modo história avançou até a criação da fase tutorial, o que viabilizou a realização de duas sessões de *playtesting* e a coleta de sugestões e opiniões

² <https://www.aseprite.org/>

relativas ao rumo a ser seguido no desenvolvimento deste modo. A continuação da implementação desse modo foi postergada para focar na implementação do modo de treino do *time attack*, dada a oportunidade de aplicar o jogo durante a Semana de Extensão 2024 da UnB no evento “Torneio de Matemática Mental 2024” descrito na seção [SEMEX 2024](#).

Com isso, na versão *competition version*, o modo história está desabilitado e seus *assets* foram removidos do projeto temporariamente, tendo em vista a diminuição do tamanho do arquivo executável do jogo, permitindo assim que o *game* fosse instalado em máquinas com poucos recursos. Em trabalhos futuros envolvendo o Math Hero, o modo história deverá ser implementado levando em consideração os requisitos iniciais do projeto, para que o propósito do *game* seja mantido.

4.2.2 Modo *Time Attack*

No modo *Time Attack* foi definido que existiriam 3 modos diferentes de jogo: normal, treinamento e competição. O modo normal contaria com todas as funcionalidades existentes no modo de jogo *time attack*, porém não seria customizável e geraria cerca de 15 operações randômicas para o jogador. Já o modo treinamento possuiria a opção de selecionar quais operações ele deseja praticar e a *seed* geradora das questões. Vale ressaltar também que nos modos citados acima o cronômetro seria desligado e os resultados não seriam salvos nem localmente nem em banco de dados próprio do *game*.

O modo competição, que é o alvo principal de estudo deste trabalho, seria customizável e rastreável, contendo as seguintes opções: seleção de tipos de operações, inserção de *seed* geradora de questões aleatórias e armazenamento de resultados em um arquivo local ao computador. Foi planejado para um trabalho futuro o armazenamento em banco de dados em nuvem, para fácil acesso e coleta dos dados, tendo em vista que a coleta de dados é dificultada pelo deslocamento até o computador no qual o *game* foi jogado.

Em relação aos *assets* utilizados no *game design* do modo *time attack*, foram obtidas inspirações imagens de telas de computadores antigos, e imagens de games que utilizam *pixel art* como: Celeste³, Shovel Knight⁴ e Fez⁵.

Dentre as três modalidades de *time attack* inicialmente planejadas (modo competição, modo normal e modo treino), apenas uma mescla entre o modo treino e o modo competição foi implementada até o momento. O modo treino compartilha com o modo competição apenas a inserção de uma *seed* para padronização dos resultados e a coleta dos dados após a finalização da resolução da sequência de expressões. As características

³ <https://www.celestegame.com/>

⁴ <https://www.yachtclubgames.com/games/shovel-knight-treasure-trove/>

⁵ <https://store.steampowered.com/app/224760/FEZ/>

não compartilhadas com o modo competição são: a ausência de penalidades para erros, a configuração livre de operações e não coletar os dados de jogo.

Para a separação completa entre os dois modos de jogo, seria necessária a detecção do parâmetro de modo de jogo para que, ao selecionar o modo treino, as penalidades sejam desabilitadas, a configuração livre de operações seja possível e a coleta dos dados de jogo seja desabilitada. Já quando o parâmetro de modo de competição for selecionado, todas essas opções devem ser habilitadas, com exceção da livre configuração de operações, tendo em vista que a configuração de operações seria feita automaticamente, de acordo com a *seed* da competição, facilitando assim a configuração do ambiente de competição e garantindo um ambiente justo e igual para todos os competidores.

4.3 Implementação

A escolha da utilização da língua inglesa para o nome do jogo teve como motivação principal a intenção do lançamento do *game* em plataformas digitais como o Itch.io e a Steam, facilitando a disseminação do jogo mundialmente.

A primeira etapa da implementação foi a criação dos *scripts* do *game* utilizando a linguagem de *scripting* própria da *game engine* escolhida, a *GDScript*. Anteriormente, foi realizado estudo de boas práticas atreladas à linguagem e à Godot por meio de vídeos educativos propostos pela comunidade e pela documentação oficial da Godot.

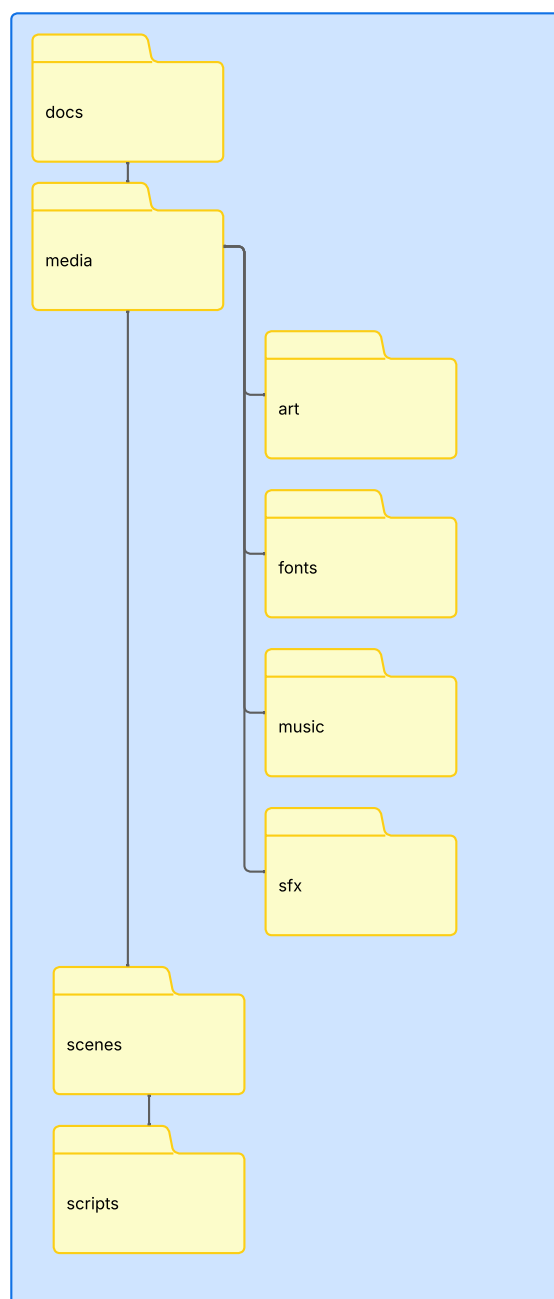
Em seguida, a próxima etapa realizada foi relacionada à implementação das mecânicas. Utilizando as boas práticas e a documentação da Godot (GODOT, 2023), foi possível realizar a criação de *scripts* específicos para a implementação de padrões arquiteturais de código, como o *singleton GameManager*, pode ser visualizado no Apêndice A.

A estrutura de organização de arquivos definida para o projeto é apresentada pela Figura 8. A pasta **docs** contém toda a documentação do game gerada à partir da ferramenta **docsify**. A pasta **media** contém 4 subdivisões relacionadas à arquivos de mídia do jogo, sendo elas: **art**, que está relacionada aos **assets** de arte criados para o *game*; **fonts**, que contém todos os arquivos de fontes necessários para as interfaces de usuário; **music**, que contém trilhas sonoras, música ambiente; e **sfx**: que contém todos os arquivos relativos aos efeitos sonoros.

A pasta **scenes** contém os arquivos com extensão ***.tscn** relativos às *scenes* das diferentes fases do *game*. Por último, a pasta **scripts** é a pasta que contém todos os *scripts* escritos em *GDScript*.

Para a versão *alpha* do *game* o modo *time attack* e seu modo de competição foram alvo do planejamento da implementação. Porém, a implementação real consistiu em fazer

Figura 8 – Estrutura de pastas Math Hero



Fonte: o Autor

a produção de um mínimo produto viável (MVP) específico para os eventos descritos na seção [SEMEX 2024](#). O modo implementado, denominado modo de treino, possui uma versão inicial daquilo que foi planejado para o modo de competição contendo a inserção de *seeds* para randomização e padronização das expressões a serem resolvidas pelo jogador, o registro de dados pertinentes ao modo de competição como: tempo decorrido, operações habilitadas e nome identificador; e a seleção da dificuldade de resolução das expressões.

A priorização do modo de treino deu-se pela emergente oportunidade de aplicação e validação do modelo e temática do jogo por meio de atividades como a *SEMEX 2024* da UnB, onde foi proposto e planejado que o *game* fosse utilizado para promover uma pequena competição matemática entre os participantes inscritos.

4.3.1 Time Attack

Para desenvolver o modo *time attack* do *game*, primeiro fez-se necessário mapear os tipos de expressões para então definir as propriedades a serem utilizadas pelo jogador ao jogar este modo. As operações definidas, bem como suas propriedades, foram a adição, a subtração, a multiplicação e a divisão. As características referentes a cada operação e suas propriedades estão descritas nas Tabelas 3, 4, 5 e 6 respectivamente.

Em todas as tabelas, foram consideradas apenas operações que geram resultados inteiros e positivos, isto é, com resultado $x \in \mathbb{Z}$ e $x \geq 0$. Na Tabela 3 estão separados os tipos de expressões de adição definidas para o modo *time attack*. Levando em consideração expressões do tipo: $a + b = x$, a coluna “Número de dígitos das parcelas” representa o número de dígitos de cada parcela a e b e a coluna “Vai um” representa a utilização, ou não, da propriedade de “vai um” que ocorre quando a soma dos dígitos de centena, dezena ou unidade é maior ou igual a 10.

Tabela 3 – Expressões disponíveis para a operação de adição

Número de dígitos das parcelas	“vai um”
1	✗
1	✓
2	✗
2	✓
3	✗
3	✓

Fonte: o Autor

Já na Tabela 4 estão dispostos os tipos escolhidos de expressões de subtração com base no modelo $a - b = x$. As colunas “Dígitos de a ” e “Dígitos de b ” representam a quantidade de dígitos de cada parcela a e b do modelo. A coluna “Propriedade” representa a propriedade utilizada nas parcelas a e b da expressão, um “-” representa que nenhuma propriedade especial foi utilizada. Por fim, a coluna “Pegar emprestado” representa a utilização, ou não, da propriedade de “pegar emprestado” ao solucionar a expressão. Essa propriedade ocorre quando a subtração de um dos dígitos de centena, dezena ou unidade representaria um resultado negativo, assim fazendo-se necessário “pegar emprestado” do dígito subsequente.

Tabela 4 – Expressões $a - b$ disponíveis para a operação de subtração

Dígitos de a	Propriedade	Dígitos de b	“pegar emprestado”
1	-	1	✗
2	Múltiplo de 10	1	✓
2	-	1	✗
2	-	1	✓
3	Múltiplo de 100	2	✓
3	-	2	✗
4	Múltiplo de 1000	3	✓
3	-	2	✓

Fonte: o Autor

Na Tabela 5 foram expostos os tipos de expressões de multiplicação escolhidos para a implementação. Usando o modelo $a \times b = x$, as colunas “Dígitos de a ” e “Dígitos de b ” representam a quantidade de dígitos em cada uma das parcelas a e b da expressão. A coluna “Propriedade” apresenta as propriedades especiais utilizadas na expressão, e um “-” informa que nenhuma propriedade especial foi utilizada. Na coluna “Vai um”, está representado a utilização, ou não, da propriedade de “vai um” na solução.

Tabela 5 – Expressões $a \times b$ disponíveis para a operação de multiplicação

Dígitos de a	Propriedade	Dígitos de b	“Vai um”
1	-	1	✗
2	$a = 11$	1	✓
2	$a = 11$	2	✗
2	$a = 11$	2	✓
2	com 5 na unidade	2	✓
2	com 5 na dezena	2	✓
3	-	2	✓
2	$a = 11$	3	✗
2	$a = 11$	3	✓
2	-	2	✓
3	-	3	✓

Fonte: o Autor

Por último, na Tabela 6 estão expostas as propriedades e o número de dígito de cada parcela de expressões do tipo: $a \div b = x$. onde a coluna “Propriedade” representa a utilização ou não de propriedades especiais na expressão.

Tabela 6 – Expressões $a \div b$ disponíveis para a operação de divisão

Dígitos de a	Propriedade	Dígitos de b
2	-	1
2	-	2
3	-	2
2	$b = 5$	1
3	-	1

Fonte: o Autor

4.3.1.1 Escolha das referências

Conforme desenvolvimento do modo de jogo *Time Attack*, foram utilizadas referências “retrô” para a criação dos *assets* necessários para a criação da tela. Uma das principais referências foi a de telas de computador monocromáticas verdes, como pode ser observado na Figura 9.

Figura 9 – Tela de PC monocromática verde



Fonte: Google Imagens

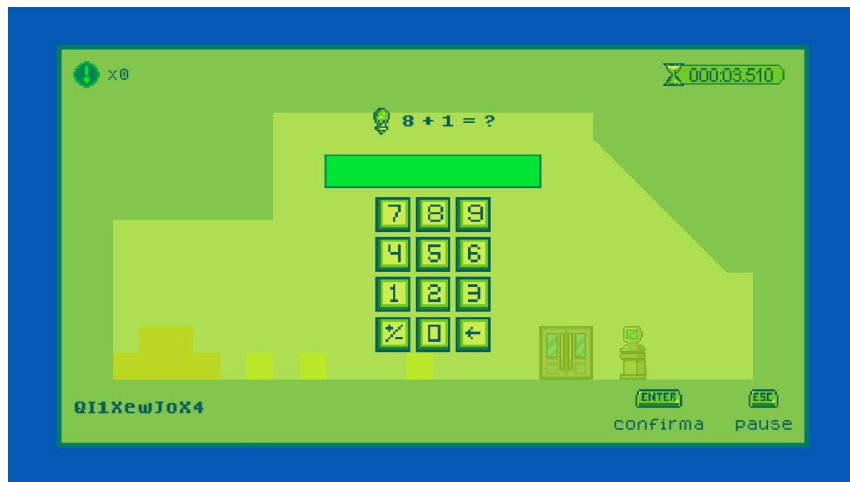
Também foi necessário selecionar a paleta de cores correspondente e realizar a criação dos *assets* no software de criação de *sprites* *Aseprite*. A paleta de cores (Figura 10) escolhida tem o nome de *Gameflower green* e foi obtida do site lospec.com.

Com a utilização dessa paleta de cores, foi possível criar os *assets* e renui-los no *tileset* representado no Apêndice *Tilemap do Modo Time Attack*. Assim, unindo as referências buscadas à paleta de cores escolhida, a interface de batalha de matemática foi evoluída para a imagem apresentada na Figura 11.

Figura 10 – Paleta de cores *Gameflower green*

Fonte: lospec.com

Figura 11 – Tela do modo Time Attack



Fonte: o Autor

O primeiro protótipo da funcionalidade de batalha de matemática, que faria parte do modo história, pode ser visto na Figura 12a. Conforme implementação do modo de treino, utilizado na *SEMEX 2024* (Figura 12b), o modo *time attack* recebeu elementos gráficos que representam o tempo decorrido, a quantidade de erros, um teclado numérico para representação visual da seleção dos números, a *seed* do jogo e as instruções visuais pertinentes ao modo, como o botão “ENTER” para confirmar o resultado da expressão e o botão “ESC” para pular a expressão atual e ir para a próxima, com uma penalidade de 30 segundos ao tempo final.

4.4 Playtesting

A fase de *playtesting* serviu como principal validadora da temática escolhida, mas também como medidor da aceitação da proposta do jogo. Nesta fase, foi importante aceitar todo tipo de *feedbacks* e propostas de melhoria passadas pelos jogadores escolhidos para teste. Foram construídas duas personas fictícias, para preservação da identidade das pessoas, para realização do *playtesting* e colheita de sugestões de melhorias, uma adolescente e uma adulta (Tabela 7), e ambas jogaram a versão *alpha* do jogo.

Figura 12 – Evolução da mecânica utilizada no *time attack*

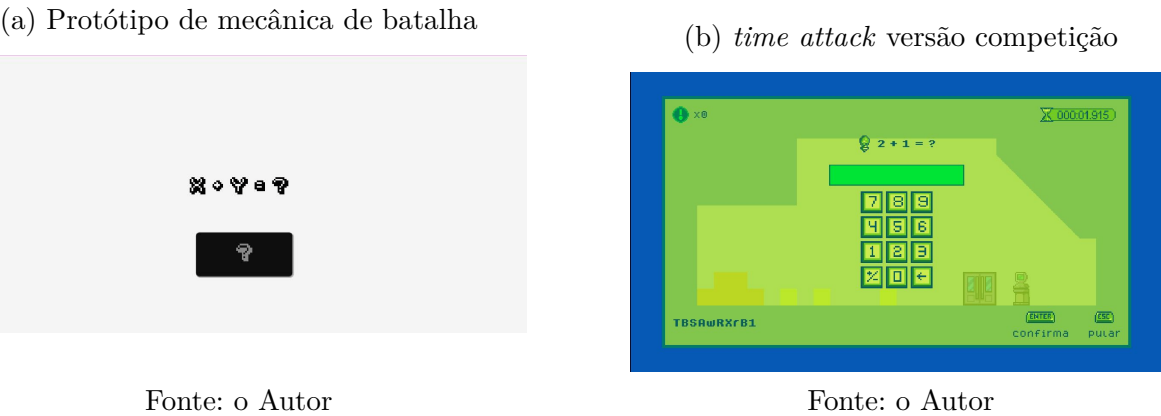
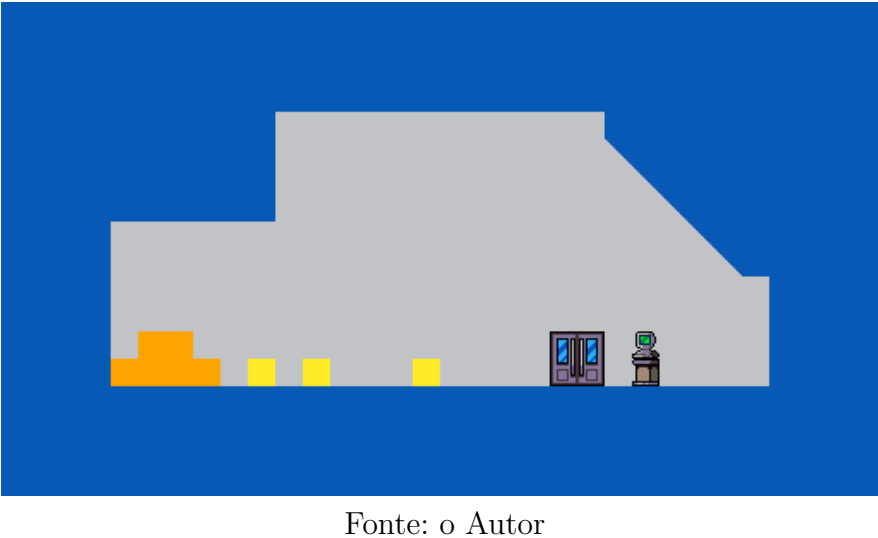


Tabela 7 – Identificação das personas escolhidas para sessão de playtesting.

Nome	Idade	Profissão
Manuela	15	Estudante
Pedro	24	Comunicador social

Fonte: o Autor

Figura 13 – *Scene* para realização de testes



Durante a sessão de *playtesting*, Manuela foi a principal fonte de coleta de dados e sugestões, visto que ela está dentro do público alvo e, portanto, traz uma visão do público alvo em relação ao produto final. A versão protótipo do game, representada pela Figura 13, possuía uma mecânica de movimentação limitada à fase em que o jogador se encontrava, além da mecânica de “encontro” com um NPC hostil, que chamava o jogador para uma batalha de matemática (Figura 12a). As sugestões dadas por Manuela e Pedro

após a sessão de *playtesting*, e as observações feitas acerca do comportamento de ambos durante a sessão, estão descritas na Tabela 8.

Tabela 8 – *Feedbacks* dados pelas pessoas escolhidas para a sessão de *playtesting*.

Nome	Sugestão	Observação
Manuela	“Achei muito entediante a batalha de matemática, seria legal se fosse tipo um <i>escape room</i> , daí a pessoa precisaria de usar as técnicas pra encontrar pistas pra abrir as portas e passar de fase.”	Jogadora iniciante mas teve facilidade na adequação aos controles. Sentiu falta de direcionamentos dados ao jogador
Pedro	“Achei boa a mecânica de batalha, mas bem lenta. Acho que poderia mudar ela e deixar tipo uma batalha com <i>timer</i> rodando.”	Jogador experiente. Sentiu falta de indicadores de progresso na fase.

Fonte: o Autor

A principal sugestão foi fazer ajustes na jogabilidade para que o game se tornasse mais dinâmico, além de realizar ajustes na forma de progressão. A decisão tomada a partir dessa sugestão foi a de realizar um ajuste no game design para que o progresso se comportasse de forma similar à um “*escape room*”, onde os jogadores precisam encontrar pistas e solucionar problemas para progredir e, eventualmente, sair da sala onde se encontram atualmente para outra com novos desafios.

Assim, foi possível planejar ajustes no *game design*, para que o jogo seja mais atrativo para o usuário final e cumpra com o principal requisito de fortalecimento do aspecto lúdico do *game*. Embora os ajustes e sugestões tenham sido levados em consideração na evolução da mecânica, elas não estão presentes no modo de treino, que foi desenvolvido como versão final deste trabalho, portanto a aplicação e coleta de novas sugestões e observações foram elencadas como tarefas para trabalhos futuros.

4.5 SEMEX 2024

O *game* Math Hero foi utilizado no evento “Torneio de Matemática mental 2024”, entre os dias 04/11/2024 e 08/11/2024, no Campus UnB Gama: Faculdade de Ciências e Tecnologias em Engenharia. Foram ofertadas 60 vagas, onde 30 destas vagas foram destinadas ao turno matutino e as outras 30 ao turno vespertino. O evento foi guiado por um conjunto de regras básicas com a intenção de promover uma competição saudável e também como forma de padronizar a jogabilidade, fazendo com que os dados coletados

fossem comparáveis. As regras estabelecidas aos competidores em relação à execução do *game* e ao comportamento de cada um dos competidores foram:

1. O torneio de matemática mental é uma competição individual, onde os participantes se enfrentam, em turnos, para decidir quem consegue resolver problemas de adição, subtração, multiplicação e divisão mais rapidamente.
2. O programa utilizado pelos participantes é o Math Hero, software livre desenvolvido pelo estudante Matheus Pimentel Leal.
3. A cada rodada os participantes tentarão resolver 20 operações. As operações são as mesmas para todos os participantes: isto é garantido pela configuração prévia do modo competição. A organização do evento determinará qual será a configuração de cada rodada, e os participantes devem seguir as configurações à risca.
4. Caso o participante, por erro, descuido ou vontade própria, use uma configuração diferente da determinada pela organização, ele será eliminado da rodada, recebendo como tempo final 100 minutos.
5. Sendo uma competição individual, não é permitido, durante uma rodada, a comunicação com outros competidores ou o uso de calculadoras, celulares, internet, dispositivos eletrônicos ou papel e caneta. A violação desta regra implicará na desclassificação do competidor, que deverá deixar a competição.
6. A cada ciclo de duas rodadas serão classificados um certo número de competidores (a quantidade será divulgada pela organização a cada início de ciclo). Os não classificados deixam a competição.
7. A cada novo ciclo o nível de dificuldade das operações aumenta, partindo de iniciante até atingir o nível difícil.
8. A cada erro cometido o competidor adicionará uma penalidade de 30 segundos ao seu tempo final. Esta penalidade é computada automaticamente pelo Math Hero.
9. Caso o competidor deseje, ele pode saltar a operação atual, pagando uma penalidade de 5 minutos no seu tempo final.
10. O casos omissos serão tratados pela coordenação do evento.

Em relação ao modelo da competição, foram definidas 3 etapas com 2 rodadas cada. Ao final da primeira rodada os 3 alunos com piores tempos foram eliminados e, ao final da segunda rodada, os 6 alunos com os piores tempos também foram eliminados. Dessa forma, na primeira rodada, pelo período da manhã, 31 alunos participaram da

primeira rodada, 28 da segunda rodada e 22 da terceira rodada. E no período da tarde, 17 competidores participaram da primeira rodada, 14 da segunda rodada e, por último, 8 participaram da terceira rodada.

O objetivo do evento “Torneio de Matemática mental 2024” era estimular a busca pelo conhecimento e desenvolvimento de técnicas facilitadoras para a realização de cálculos utilizando somente o fator mental, sem o auxílio de papéis, calculadoras ou quaisquer dispositivos que auxiliam na resolução de equações matemáticas.

Os alunos de uma escola de ensino médio do Gama foram convidados a participar do evento durante o período da manhã, como forma de incentivar e testar as habilidades dos mesmos em um ambiente lúdico. As técnicas de matemática mental apresentadas na Seção [Matemática mental](#) do Capítulo [Referencial Teórico](#) foram previamente ensinadas aos alunos com o auxílio de uma versão de testes do Math Hero. No total, 52 alunos participaram da competição. O resultado da competição na parte da manhã com os alunos de ensino médio pode ser observado por meio dos Tabela 9, que apresenta um “top 10” alunos ranqueados em ordem crescente pelo tempo total para resolução das expressões, no formato: *minutos : segundos : milissegundos*.

Tabela 9 – Quadro de líderes de alunos de ensino básico treinados com as técnicas de matemática mental

Posição	Nome	Tempo Total
1	Aluna A	005:03.564
2	Aluna B	005:10.974
3	Aluno C	005:35.395
4	Aluna D	005:35.600
5	Aluno E	007:12.161
6	Aluno F	009:51.062
7	Aluna G	010:18.884
8	Aluno H	011:19.248
9	Aluno I	011:59.204
10	Aluna J	012:31.304

Fonte: o Autor

Como parâmetro de comparação, na parte da tarde a competição foi aberta ao público geral universitário, sem treinamento prévio de técnicas de matemática mental e sob as mesmas regras e estruturas utilizadas no período da manhã, para que fosse possível a análise do desempenho de pessoas sem instrução prévia sobre as técnicas de resolução mental. No período da tarde, no total compareceram 17 estudantes

Como podemos perceber, o desempenho dos alunos de ensino médio foi superior, considerando apenas o “top 5” de participantes no período da tarde e no período da manhã. O tempo médio para finalização das questões propostas dos “top 5” participantes

Tabela 10 – Quadro de líderes de alunos de ensino superior sem treinamento prévio com as técnicas de matemática mental

Posição	Nome	Tempo Total
1	Competidor A	005:15.583
2	Competidor B	010:31.421
3	Competidora C	010:55.652
4	Competidor D	011:04.054
5	Competidora E	014:50.528
6	Competidor F	015:28.875
7	Competidor G	015:52.297
8	Competidor H	070:00.000

Fonte: o Autor

do período da manhã foi de: $5min43s539ms$. Já o tempo médio dos “top 5” participantes do período da tarde foi de: $10min31s447ms$. Esta comparação sobre o desempenho de pessoas com e sem o conhecimento prévio reforça a validade da utilização de técnicas de matemática mental para a obtenção de resultados mais rápidos e corretos para expressões matemáticas.

O *game* Math Hero teve um desempenho aceitável durante a realização do evento “Torneio de matemática mental 2024”. Porém, algumas dificuldades foram encontradas, como a dificuldade de inserção das *seeds* por parte de alguns alunos, fazendo com que alguns registros não fossem captados. Essa falha se deu por conta da funcionalidade de inserção manual das *seeds*.

Outro fato relevante foi a seleção de uma versão desatualizada do *game* por parte de um estudante da turma de ensino médio, que foi o público do período da manhã. Esse fato se deu pela existência de uma versão do jogo instalada nos computadores do laboratório utilizado e de uma versão com atualizações de última hora que foi colocada na área de trabalho dos computadores do laboratório.

De forma crítica, o resultado obtido pela utilização do jogo no evento foi satisfatório, mas com ressalvas. Melhorias nas funcionalidades do modo de treino e a criação do modo dedicado exclusivamente a competições foram as sugestões levantadas para realização de trabalhos futuros significativos para a evolução do software, além de proporcionar a realização de eventos com maior suavidade e menos dificuldades.

4.6 Análise pós-implementação

A análise pós-implementação foi feita para que os pontos de dificuldade encontrados durante a utilização do *game* Math Hero em competições de matemática mental

tenham suas motivações encontradas e sanadas.

Na implementação atual do jogo, o modo implementado (modo *time attack*) possui apenas uma modalidade, oriunda de uma mescla entre as modalidades de treino e de competição, utilizando características planejadas para ambos os modos. A separação completa entre as modalidades de competição e de treino inclui a adição de mais uma configuração de modo *time attack*, que consiste em habilitar o cadastro prévio de *seeds* de competição em um servidor remoto, para que, ao informar o código da competição como, por exemplo, “SEMEX2024”, a *seed* do jogo, juntamente com suas configurações de rodadas, sejam automaticamente carregadas, facilitando assim a configuração e utilização em eventos futuros. Esta funcionalidade não foi implementada devido a falta de organização prévia, dada urgência de utilização do *game* no evento “Torneio de matemática mental”. Com isso, optou-se por utilizar uma mescla entre as modalidades de treino e de competição levando em consideração a facilidade de implementação e a rapidez na entrega de um mínimo produto viável.

Outra funcionalidade de suma importância para a utilização do *software* em competições é a adição de um quadro de líderes que realize a exposição dos resultados, ordenados do menor tempo total até o maior tempo total. Este quadro de líderes teria facilitaria a organização e traria rapidez na divulgação dos resultados da rodada, bem como dos resultados finais.

Para melhorar consideravelmente a mão de obra necessária para realização de eventos, como o “Torneio de matemática mental 2024”, é importante que seja implementado um cadastro de *seeds* e de configurações de operações (habilitar ou não certas operações). Esse cadastro consiste em armazenar, remotamente, uma chave única para cada competição, bem como suas configurações, sendo elas: o número de rodadas, o tempo limite e as *seeds* utilizadas em cada rodada.

Finalmente, para aumentar o alcance do jogo e fazer com que cada vez mais alunos do ensino básico e superior tenham acesso ao *game*, será necessário exportar o jogo para plataformas como o Android e o iOS. Esta funcionalidade não foi implementada devido a falta de conhecimento prévio relacionado à publicação, aprovação e manutenção do software em lojas de aplicativos *mobile*.

5 Considerações Finais

Este trabalho propôs um jogo do gênero educacional chamado Math Hero, com o objetivo de introduzir técnicas de cálculo mental para alunos do ensino básico brasileiro por meio de uma abordagem lúdica, para que o estudante possa aprender de forma criativa e leve. Ao desenvolver um *serious game* que não deixa o aspecto lúdico de lado, os jogadores poderão sentir-se em uma atividade recreativa ao mesmo tempo que aprendem por meio do aspecto educacional.

Com a utilização das práticas recomendadas pela *game software engineering* (GSE), o *game* foi desenvolvido para ser relevante no meio pedagógico para o ensino de matemática. Os fatores de principal relevância foram a existência de modos de jogo que comportem tanto o ensino de técnicas de matemática mental quanto a utilização do jogo em competições que estimulem o aprendizado.

A dificuldade do balanceamento entre os aspectos lúdicos e educacionais foi notável durante o desenvolvimento do game. Percebeu-se que a introdução do aspecto educacional em um *game* causa resistência por parte do público alvo por gerar sensação de obrigação relacionada ao desempenho. Por meio de sessões de *playtesting* foi possível coletar *feedbacks* para planejar ajustes no design do jogo para reduzir a sensação de obrigação e trazer divertimento aos jogadores durante o jogo no modo história. Os ajustes programados na realização de trabalhos futuros foram a exclusão da mecânica de batalha de matemática e a inclusão de obstáculos para fazer o *game* ser como um “*escape room*”.

Outra dificuldade encontrada foi a escolha dos *assets* do jogo. Esta escolha deve ser feita em concordância com os aspectos definidos no *game design*, e por isso, muitas vezes os desenvolvedores optam por realizar a criação de *assets* personalizados para cada game. Porém, com a restrição de tempo do projeto, não foi possível realizar a criação personalizada dos *assets* para a solução visto a necessidade da validação do *game design* proposto. Durante a procura de *assets* personalizados, a barreira do preço de *assets* disponibilizados no *website ict.io* foi outra dificuldade que precisou ser superada.

A utilização da GSE foi um fator de sucesso no desenvolvimento do *game* pois proporcionou a padronização de métodos e escolha de ferramentas adequadas para a solução. Tendo em vista o caráter iterativo da GSE, foi possível realizar validações e ajustes constantes no comportamento do jogo.

A escolha de ferramentas alinhadas com os requisitos definidos inicialmente trouxe facilidade de desenvolvimento por proporcionar padronização de configuração entre máquinas de trabalho com diferentes sistemas operacionais.

5.1 Análise dos objetivos

Considerando os objetivos específicos expostos na seção [Objetivos](#) da [Introdução](#), a implementação do jogo Math Hero satisfaz os objetivos específicos:

- aplicar padrões de *game design* atrativos para o público alvo;
- coletar, no mínimo, os seguintes dados de jogabilidade do modo competitivo: tempo decorrido até a finalização, nome do jogador, configuração das questões e *seed* de jogo; e
- analisar os resultados obtidos a partir dos dados de jogabilidade dos jogadores.

Padrões de *game design* foram aplicados por meio da utilização de referências modernas para construção dos *assets* utilizados na implementação. Os dados de tempo decorrido até a finalização, nome do jogador, configuração das questões e da *seed* do jogo foram coletados e armazenados localmente em um diretório específico do *game*, criado automaticamente durante a finalização das questões e inserção do nome do jogador.

Já os resultados obtidos por meio de sessões de *playtesting* foram analisados e levados em consideração para o planejamento e implementação de funcionalidades futuras e melhorias na execução e jogabilidade do Math Hero.

Por fim, os objetivos específicos não satisfeitos em sua totalidade foram:

- desenvolver um modo de jogo que cativa o jogador por meio de elementos narrativos a aprender técnicas de matemática mental; e
- adicionar um modo de jogo competitivo que estimule o aprendizado e a competição saudável.

O desenvolvimento de um modo de jogo que funcionasse como um “modo história” não foi realizado para ceder espaço para o desenvolvimento de um mínimo produto viável que atendesse as necessidades do evento “Torneio de matemática mental” realizado na Semana de Extensão Universitária 2024 da UnB, dada a grande oportunidade de coletar dados que suportem o ensino de técnicas de matemática mental no ensino básico. E, pelo mesmo motivo, a configuração de competição do modo *time attack* não foi implementada em sua totalidade, dando espaço para uma mescla entre a configuração de treino e de competição.

5.2 Trabalhos futuros

Um dos mais importantes trabalhos futuros planejados é a continuidade do desenvolvimento do *Math Hero*, além da liberação das próximas versões do *game*, que contarão com o modo história completo e refatorado, um quadro de líderes para modo *time attack*, além de novos modos, sendo estes um modo de treino e um modo normal sem a postagem do tempo no quadro de líderes. Também estão sendo planejadas versões do jogo para plataformas *mobile* e *web*, com o intuito de ampliar o alcance do *game*.

A aplicação eficaz das sugestões levantadas até o momento possibilitará a evolução constante do projeto, bem como a sua utilização em diferentes contextos. Uma melhoria significativa seria a implementação completa do modo história do jogo, utilizando a temática de “*escape room*” para validação e realização de novas sessões de coleta de *feedbacks* através do *playtesting*.

Referências

- ARJORANTA, J. How to Define Games and Why We Need to. *The Computer Games Journal*, v. 8, n. 3, p. 109–120, dez. 2019. ISSN 2052-773X. Disponível em: <<https://doi.org/10.1007/s40869-019-00080-6>>. Citado na página 18.
- BENJAMIN, A.; SHERMER, M. *Secrets of Mental Math: The Mathemagician's Guide to Lightning Calculation and Amazing Math Tricks*. [S.l.]: Three Rivers Press, 2006. Google-Books-ID: aaT_KKERF6AC. ISBN 978-0-307-33840-2. Citado 6 vezes nas páginas 24, 25, 26, 29, 31 e 32.
- BERTICELLI, D. D.; ZANCAN, S. Aritmética com cálculo mental: “como você fez?”. *Revista de História da Educação Matemática (Online)*, v. 9, 2023. ISSN 2447-6447. Publisher: SBHMat. Citado 2 vezes nas páginas 16 e 24.
- BITTNER, K.; SPENCE, I. *Use Case Modeling*. Addison Wesley, 2003. (Addison-Wesley object technology series). ISBN 9780201709131. Disponível em: <<https://books.google.com.br/books?id=zvxfXvEcQjUC>>. Citado na página 22.
- BRACKEYS. *Basic Principles of Game Design*. 2007. Disponível em: <<https://www.youtube.com/watch?v=G8AT01tuyrk>>. Citado na página 39.
- CARMONA, O. T. *Game development - From the vision to the final product*. Tese (Bachelor thesis) — Universitat Politècnica de Catalunya, fev. 2016. Accepted: 2016-02-25T08:21:00Z. Disponível em: <<https://upcommons.upc.edu/handle/2117/83419>>. Citado na página 23.
- CHUECA, J. et al. The consolidation of game software engineering: A systematic literature review of software engineering for industry-scale computer games. *Information and Software Technology*, v. 165, p. 107330, jan. 2024. ISSN 0950-5849. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0950584923001854>>. Citado na página 19.
- COSTA, J. S. C. d. Cálculo mental nos anos iniciais do ensino fundamental. *Anais do Seminário Sul-Mato-Grossense de Pesquisa em Educação Matemática*, v. 12, n. 1, set. 2018. ISSN 2448-2943. Number: 1. Disponível em: <<https://periodicos.ufms.br/index.php/sesemat/article/view/6010>>. Citado na página 16.
- EMMEL, R.; COSTA, P. d. *Vista do O Ensino da Matemática, a aprendizagem e o fracasso escolar: uma análise dessas relações no Ensino Médio Integrado de uma instituição da rede federal de Ensino Básico, Técnico e Tecnológico*. 2019. Disponível em: <<https://periodicos.ifrs.edu.br/index.php/REMAT/article/view/3356/2276>>. Citado na página 15.
- GODOT. *Godot Docs – 4.1 branch*. 2023. Disponível em: <<https://docs.godotengine.org/en/stable/index.html>>. Citado na página 43.
- HERMANN, W.; COQUEIRO, V. d. S.; PACHECO, P. S. Um panorama das publicações de quatro periódicos da área de educação de matemática a respeito de jogos como

recursos didáticos para o ensino de matemática. *Research, Society and Development*, v. 9, n. 10, p. e6639109002–e6639109002, out. 2020. ISSN 2525-3409. Number: 10. Disponível em: <<https://rsdjournal.org/index.php/rsd/article/view/9002>>. Citado na página 15.

HUANG, Q. *Game Theory*. intechopen, 2010. ISBN 978-953-307-132-9. Disponível em: <<https://www.intechopen.com/books/3778>>. Citado na página 19.

IREB. *Downloads - IREB*. 2022. Disponível em: <<https://www.ireb.org/en/downloads/>>. Citado na página 20.

LUCKESI, C. Ludicidade e formação do educador. *Revista Entreideias: educação, cultura e sociedade*, v. 3, n. 2, set. 2014. ISSN 2317-1219. Number: 2. Disponível em: <<https://periodicos.ufba.br/index.php/entreideias/article/view/9168>>. Citado na página 15.

Mark Brown - GMTK. *Valve's "Secret Weapon"*. 2023. Disponível em: <<https://www.youtube.com/watch?v=9Yomqk0C6kE>>. Citado na página 23.

MORAES, E. A. D. *Utilização de uma Estratégia para Identificação de Fontes de Informação na Fase de Elicitação*. Tese (Mestre em Informática) — Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, Brazil, maio 2008. Disponível em: <http://www.maxwell.vrac.puc-rio.br/Busca_etds.php?strSecao=resultado&nrSeq=15760@1>. Citado na página 21.

NASCIMENTO, D.; LARA, V. M.; TRUQUETE, M. K. Os Jogos Didáticos Como Recurso Metodológico Para o Ensino de Matemática no Ensino Fundamental. *Revista Exitus*, v. 13, p. e023017–e023017, fev. 2023. ISSN 2237-9460. Disponível em: <<https://portaldeperiodicos.ufopa.edu.br/index.php/revistaexitus/article/view/1965>>. Citado 2 vezes nas páginas 15 e 24.

OCDE. *PISA 2022 results: Preparing students for a changing world*. 2023. Disponível em: <<https://www.oecd.org/publication/pisa-2022-results/>>. Citado na página 16.

PAGOTTO, T. et al. Scrum solo: Software process for individual development. *2016 11th Iberian Conference on Information Systems and Technologies (CISTI)*, p. 1–6, 2016. Citado na página 35.

PEDERSEN, R. E. *Game Design Foundations*. [S.l.]: Wordware Publishing, Inc., 2003. Google-Books-ID: DvYmFr_uaWwC. ISBN 978-1-55622-973-2. Citado na página 22.

ROGERS, S. *Level Up!: The Guide to Great Video Game Design*. [S.l.]: John Wiley & Sons, 2010. Google-Books-ID: 8w_ETFmHrewC. ISBN 978-0-470-97092-8. Citado 3 vezes nas páginas 18, 19 e 22.

SHELL, J. *The Art of Game Design: A book of lenses*. 1. ed. London: CRC Press, 2008. Publication Title: The Art of Game Design. ISBN 0-12-369496-5. Citado na página 19.

SILVA, F. B. d.; GUALANDI, J. H.; SANTOS, P. d. O uso de jogos matemáticos no trabalho com o cálculo mental. *TANGRAM - Revista de Educação Matemática*, v. 3, n. 3, p. 60–84, set. 2020. ISSN 2595-0967. Number: 3. Disponível em: <<https://ojs.ufgd.edu.br/index.php/tangram/article/view/12264>>. Citado na página 16.

SOMMERVILLE, I. *Software Engineering Global Edition*. 10th ed. edição. ed. Boston Columbus Indianapolis New York San Francisco Hoboken Amsterdam Cape Town Dubai London Madrid Milan Munich Paris Montreal Toronto Delhi Mexico City São Paulo Sydney Hong Kong Seoul Singapore Taipei Tokyo: Pearson, 2015. ISBN 978-1-292-09613-1. Citado na página 19.

SWIFT, K. *IGS 2007: Kim Swift - 'Our Journey From Narbacular Drop To Portal*. 2007. Disponível em: <<https://www.youtube.com/watch?v=2F0SVZA9flo>>. Citado na página 23.

TROYER, O. D.; JANSSENS, E. Supporting the requirement analysis phase for the development of serious games for children. *International Journal of Child-Computer Interaction*, v. 2, n. 2, p. 76–84, maio 2014. ISSN 22128689. Disponível em: <<https://linkinghub.elsevier.com/retrieve/pii/S2212868914000099>>. Citado na página 16.

WELLS, D. *Extreme Programming: A Gentle Introduction*. 2009. Disponível em: <<http://www.extremeprogramming.org/>>. Citado na página 35.

YOUTUBE. *You Tube*. 2023. Disponível em: <<https://www.youtube.com/>>. Citado na página 36.

APÊNDICE A – Implementação do *Singleton GameManager*

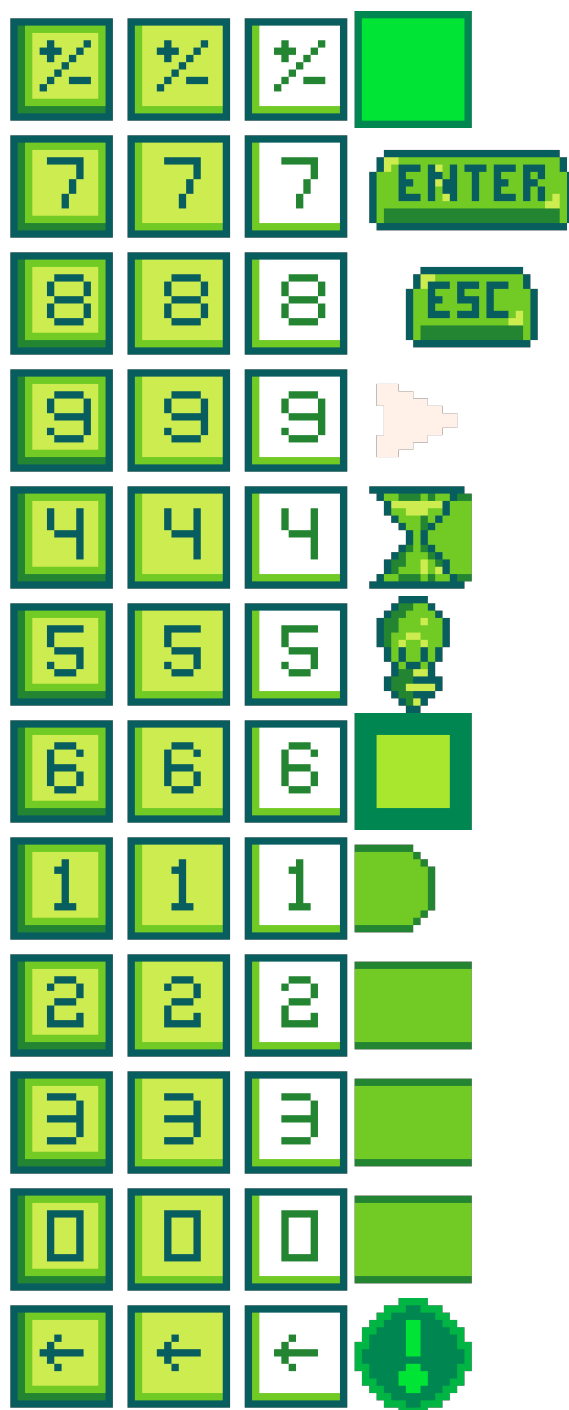
```

1  extends Node
2
3  enum Game_Modes {Story=1, TimeAttack=2, Menu=0}
4  enum Time_Attack_Modes {Competition=0, Practice=1, Normal=3}
5
6  var game_mode: int = Game_Modes.Menu
7  var time_limit: float = 0.0
8  var skip_count: int = 0
9  var tattack_options: Dictionary = {}
10 var tattack_results: Dictionary = {}
11
12 @onready var scene_tree = get_tree()
13
14 # Called when the node enters the scene tree for the first time.
15 func _ready():
16     pass
17
18 func change_scene(target_scene):
19     if not target_scene:
20         #print_debug("Target scene is null!")
21
22         scene_tree.quit() # Quit game options was pressed
23     else:
24         #print_debug(target_scene)
25         scene_tree.change_scene_to_file(target_scene)

```

APÊNDICE B – *Tilemap* do modo *time attack*

Figura 14 – *Tilemap* de artes para construção visual do modo *time attack*



Fonte: o Autor