

CENTRO UNIVERSITÁRIO SAGRADO CORAÇÃO

**DESENVOLVIMENTO DE JOGO PARA ENSINO DAS OPERAÇÕES
MATEMÁTICAS BÁSICAS PARA CRIANÇAS COM AUTISMO**

BAURU

2021

MARIA FERNANDA PIM OZAI

**DESENVOLVIMENTO DE JOGO PARA ENSINO DAS OPERAÇÕES
MATEMÁTICAS BÁSICAS PARA CRIANÇAS COM AUTISMO**

Monografia de iniciação científica
apresentada a Pró Reitoria de Pesquisa
e Pós-Graduação como parte dos pré-
requisitos para aprovação do conselho,
sob orientação do professor Me.
Patrick Pedreira Silva.

BAURU

2021

Dados Internacionais de Catalogação na Publicação (CIP) de acordo com
ISBD

O99d	Ozai, Maria Fernanda Pim Desenvolvimento de jogo para ensino das operações matemáticas básicas para crianças com Autismo / Maria Fernanda Pim Ozai. -- 2021. 43f. : il. Orientador: Prof. M.e Patrick Pedreira Silva Monografia (Iniciação Científica em Ciência da Computação) - Centro Universitário Sagrado Coração - UNISAGRADO - Bauru - SP 1. jogo. 2. interface. 3. matemática. 4. crianças autistas. I. Silva, Patrick Pedreira. II. Título.
------	---

RESUMO

A educação tem sido cada vez mais inclusiva, todavia nem sempre os recursos necessários estão disponíveis. Entre os grupos de inclusão estão às crianças autistas, que em muitos casos não conseguem aprender o conteúdo escolar por possuir algumas dificuldades que diferem o modo como às mesmas aprendem.

Muitas vezes, o educador não sabe a melhor forma de ensinar o conteúdo para essas crianças – não possuindo métodos ou recursos para desempenhar sua função corretamente. Uma das disciplinas que podem ser trabalhadas para essas crianças é a matemática, já que para ensiná-la (e aprendê-la) se faz necessário um processo de abstração, que é uma das maiores dificuldades dos autistas quanto à aprendizagem.

Pensando nisso, este projeto propõe o desenvolvimento de um software interativo (jogo) que tem como função auxiliar professores e pais no processo de aprendizagem das operações matemáticas básicas de crianças com autismo. O jogo vai trabalhar os conceitos matemáticos de forma concreta, utilizando uma interface simples e atrativa para capturar a atenção das crianças para o conteúdo que será ensinado.

Palavras-chave: jogo; interface; matemática; crianças autistas.

SUMÁRIO

1. RELATO DA PESQUISA.....	6
2. INTRODUÇÃO.....	7
3. OBJETIVOS.....	9
3.1. OBJETIVO GERAL.....	9
3.2. OBJETIVOS ESPECÍFICOS.....	9
4. REFERENCIAL TEÓRICO.....	10
4.1. JOGOS EDUCATIVOS.....	10
4.2. OPERAÇÕES BÁSICAS.....	11
4.3. INTERFACE GRÁFICA.....	12
4.4. TRANSTORNO ESPECTRO AUTISTA (TEA).....	13
4.5. ENGENHARIA DE SOFTWARE.....	14
4.5.1. TESTE DE SOFTWARE.....	16
4.5.2. NIVEIS DE TESTE DE SOFTWARE.....	19
4.5.3. TECNICAS DE TESTE DE SOFTWARE.....	23
5. METODOLOGIA.....	25
5.1. LEVANTAMENTO BIBLIOGRÁFICO.....	25
5.2. LINGUAGEM DE PROGRAMAÇÃO.....	25
5.3. DEFINIÇÃO DAS ATIVIDADES.....	26
5.4. JOGOS DIGITAIS.....	26
5.5. INTERFACE.....	26
6. RESULTADOS.....	28
6.1. TECNOLOGIAS E FERRAMENTAS UTILIZADAS.....	28
6.2. FUNCIONAMENTO DO JOGO.....	28
7. CONSIDERAÇÕES FINAIS.....	37
8. REFERÊNCIAS.....	38

1 RELATO DA PESQUISA

Com a aprovação do projeto, iniciou-se pesquisas relacionadas ao aprendizado de crianças autistas: formas de compreensão, didáticas, maneiras de apresentar o conteúdo, além de psicologia das cores. Também foi efetuada uma intensa pesquisa relacionada as principais tecnologias que poderiam melhor atender as necessidades do projeto: frameworks de desenvolvimento web e linguagens de programação.

Dando seguimento a pesquisa, primeiramente foi desenvolvido a dinâmica e conteúdo didático de cada fase, divididas em duas etapas: elaboração de enunciado (didáticas) e elaboração das fases (design, definição de cores e layout).

Partindo da etapa anterior, foi possível realizar a construção das fases em sua totalidade. Para isso, foi utilizado um software para edição de imagens vetoriais que possibilitou arquitetar os personagens, cenários, itens, entre outros componentes importantes para o desenvolvimento das fases.

Após as etapas citadas acima, foi iniciado uma implementação parcial (com fins de estudo) para poder definir efetivamente qual tecnologia poderá ser mais acessível e adequada ao projeto.

O desenvolvimento efetivo do jogo será iniciado a partir do mês de março, e será dividido em três etapas: estruturação e construção do layout e das regras aplicadas em cada tela (front-end), estruturação e construção de uma API (back-end), além da modelagem, estruturação e construção do banco de dados.

Com o sistema finalizado, cada funcionalidade foi testada e avaliada de acordo com a proposta inicial do projeto. As informações obtidas através deste teste serão alocadas e discutidas neste relatório.

Após as etapas acima, será iniciado os preparativos finais para entrega do projeto e apresentação no Fórum de Iniciação Científica.

2 INTRODUÇÃO

Muito se tem comentado acerca da inclusão escolar de crianças com transtorno espectro autista (TEA). Porém, em muitos casos as escolas não possuem toda a infraestrutura necessária para que essas crianças tenham um aprendizado adequado.

A tecnologia tem-se mostrado muito positiva para esse processo de inclusão, como foi proferido por Dr. Clay Brites (*apud* Adriana Del Ré, 2018):

“A tecnologia pode auxiliar no dia a dia por meio de softwares que ajudem quem tem autismo a conseguir cumprir tarefas sociais, entender linguagem não verbal, serem alfabetizados, aumentar a motivação ou interesse por atividades pedagógicas ou acadêmicas [...]”.

Segundo a compreensão de Artur Ziviani (*apud* Marcos Prudencio, 2018), o uso da tecnologia vem transformando a forma como as pessoas se relacionam e se comunicam nos vários aspectos de suas vidas – tanto familiar quanto profissional. Considerando seu crescente uso, a tecnologia torna-se um recurso indispensável nas mais diversas áreas, sendo base de conhecimento e ferramenta de trabalho para vários profissionais.

Não sendo um recurso descartado por pedagogos e profissionais da área, a modernização vem servindo como apoio ao aprendizado mais dinâmico. Como consequência, é possível manter o interesse e atenção das crianças em atividades pedagógicas.

Um recurso tecnológico que se mostra interessante – principalmente ao ser utilizado com crianças – são os jogos eletrônicos. De acordo com o entendimento de KASHIWAKURA (2008), os jogos auxiliam de forma poderosa no desenvolvimento cognitivo dos indivíduos, fornecendo-os conhecimentos sobre vários pontos vista – uma vez que, trabalhar o ensino através de meios lúdicos geram ambientes mais atraentes e gratificantes, contribuindo para o desenvolvimento integral do indivíduo.

Compreendendo isso, é possível notar os reflexos positivos que a educação aliada aos jogos pode ter na vida de uma pessoa diante de

uma sociedade cada vez mais rígida e exigente, como comentado por Isabel Cristina Machado de Lara (2003, p.2):

“[...] novas habilidades passam a ser exigidas não só no mercado de trabalho como, também, na vida social dos cidadãos. Efeito disso, a capacidade de resolver problemas, utilizar a imaginação e a criatividade passam a ser requisitos cada vez mais indispensáveis.”

Dessa forma, como instrumento para auxiliar na educação dos portadores de autismo, a proposta desse projeto é o desenvolvimento de um software interativo (jogo) que busca atender as necessidades de crianças autistas com dificuldade em aprender as operações matemáticas básicas (adição, subtração, multiplicação e divisão).

A partir daí, considera-se como objetivo dessa pesquisa enfatizar o uso da tecnologia como meio de obtenção de conhecimento, além da utilização positiva de recursos audiovisuais como um estímulo ao aprendizado.

3 OBJETIVOS

Serão apresentados os objetivos gerais e específicos desse projeto.

3.1 OBJETIVO GERAL

Desenvolvimento de um software interativo (jogo) que tem a finalidade de auxiliar crianças com autismo a aprender conceitos básicos de matemática.

3.2 OBJETIVOS ESPECÍFICOS

- Realizar estudo sobre linguagem de programação;
- Pesquisar sobre desenvolvimento de jogos digitais;
- Estudar métodos pedagógicos de ensino da matemática;
- Desenvolver uma interface simples e atrativa para crianças, sobretudo, as autistas;
- Estudar sobre desenvolvimento de softwares educativos;
- Planejar atividades que serão implementadas no jogo;
- Promover a divulgação da experiência e resultados obtidos em eventos técnicos e científicos, publicações correlatas, e participar do Congresso Anual de Iniciação Científica e Desenvolvimento Tecnológico e Inovação do Unisagrado.

4 REFERENCIAL TEÓRICO

4.1 JOGOS EDUCATIVOS

Segundo os Parâmetros Curriculares Nacionais – PCNs (*apud* Lara, 2003):

“À medida que vamos nos integrando ao que se denomina uma sociedade da informação crescente e globalizada, é importante que a Educação se volte para o desenvolvimento das capacidades de comunicação, de resolver problemas, de tomar decisões, de fazer inferências, de criar, de aperfeiçoar conhecimentos e valores, de trabalhar cooperativamente.”

Ao classificar a sociedade atual como a “sociedade da informação”, evidencia-se também a importância da tecnologia, considerando que um mundo globalizado e com acesso total a informação só foi possível através do avanço da mesma. Nesse contexto, torna-se impossível não falar dos benefícios que a tecnologia trouxe para a educação.

Uma das maneiras de integrar desenvolvimento, tecnologia e educação de crianças são os jogos eletrônicos, que se tornam uma ferramenta muito vantajosa ao aprendizado. Como observado por Lara (2003, p.2):

“[...] o jogo passa a ser visto como um agente cognitivo que auxilia o aluno a agir livremente sobre suas ações e decisões fazendo com que ele desenvolva além do conhecimento matemático também a linguagem [...]”.

Portanto, o lúdico torna-se dinâmico e faz com que os alunos sintam maior prazer em aprender. Isso resulta em uma maior identificação e imersão das crianças com a aula (Juliana Bortolucci Olivério, 2020).

4.2 OPERAÇÕES BÁSICA

É notável a importância da matemática na vida de um indivíduo, estando presente nas mais variadas atividades e contribuindo para o desenvolvimento do raciocínio lógico, criatividade e capacidade de resolver problemas (Juliana Bortolucci Olivério, 2020).

Tais características se mostram muito importantes para o profissional e o pessoal de cada indivíduo. Dessa forma, para desenvolvê-las é necessário construir o conhecimento matemático desde a infância.

De acordo com Carina Silvana Virgulino (2014):

“[...] a Matemática ocupa um lugar de destaque. Numerosas pesquisas têm apontado a relevância do trabalho com essa disciplina para as crianças pequenas, especialmente no que diz respeito à construção do conceito de número, além das noções ligadas às grandezas e medidas, bem como espaço e forma”.

Entre os vários conceitos matemáticos as operações básicas (adição, subtração, multiplicação e divisão) ganham destaque, já que elas estão constantemente presentes em nosso cotidiano. Então, considerando que desde o nascimento a criança está imersa em um universo cercado de conhecimentos matemáticos, a educação básica se torna fundamental ao desenvolvimento do aluno (VIRGULINO, 2014).

4.3 INTERFACE GRÁFICA

Ao desenvolver um software é necessário projetar uma interface gráfica de maneira atenciosa, tendo em mente as características principais dos determinados usuários.

Em concordância com Padovani e Moura (*apud* Carusi e Mont’Alvão, 2012), é importante ressaltar que as áreas onde se pode interagir (clicar) em uma interface de software devem ser representadas para auxiliar na navegação do sistema. Em contrapartida, se essas áreas não forem bem trabalhadas, poderão causar ao usuário a sensação de estar perdido.

Ao considerar que o software terá como público alvo crianças com autismo, a interface gráfica deverá ter características que sejam atrativas para esse público, motivando a utilização do software.

Uma característica importante que deve conter em um software infantil é uma distribuição simples das áreas clicáveis e seções, uma vez que as crianças precisam mentalizar as saídas das seções e as áreas interativas que levaram a ela (Carusi e Mont'Alvão, 2012).

Trabalhar corretamente cores e sons também é importante na construção de uma interface. Como observado por Ana Maria Berehoffs (2009):

“Algumas crianças possuem hipersensibilidade sonora mais aguçada que outras, além de dificuldades motoras e visuais.”

Pensando nessas dificuldades, é necessário projetar uma interface que seja adaptável para cada criança.

4.4 TRANSTORNO ESPECTRO AUTISTA (TEA)

Segundo a Organização Pan-Americana de Saúde (2017), uma a cada 160 crianças possuem autismo. Entre as principais dificuldades dessas crianças estão: o comportamento social, comunicação e linguagem. Isso resulta em um indivíduo muito voltado para si mesmo, se envolvendo em atividades individuais e repetitivas específicas que despertam seu interesse.

De acordo com Dr. Drauzio Varella [201-?], o autismo pode ser classificado em três tipos:

- **Autismo Clássico:** os indivíduos portadores desse tipo de autismo são normalmente mais voltados para si mesmo, não estabelecem contato visual e não costumam utilizar a fala como instrumento de comunicação.
- **Autismo de Alto desempenho (Asperger):** apresentam as mesmas dificuldades do autismo clássico (de forma reduzida). São inteligentes e verbais, se destacando em suas especialidades.
- **Distúrbio global do desenvolvimento sem outra especificação (DGD-SOE):** possuem dificuldade de
- interação social, porém os sintomas não são suficiente para

se encaixar no autismo clássico (torna o diagnóstico mais difícil).

Para lidar melhor com essas dificuldades é necessário um acompanhamento especializado, mas sem sempre isso é possível – como foi proferido por Kartik Mahajan (*apud* Vinicius Szafran, 2020):

"Os terapeutas humanos são cruciais, mas nem sempre estão disponíveis ou acessíveis para as famílias".

Pensando nisso, pode-se aderir a tecnologia acessível como uma aliada para os indivíduos que possuem essa condição. Um exemplo é o projeto Kiwi – um robô que utiliza inteligência artificial para contribuir a educação de crianças com autismo – que se mostrou extremamente positivo, identificando os interesses das crianças e auxiliando em suas dificuldades.

4.5 ENGENHARIA DE SOFTWARE

Engenharia de Software é o processo de estudar, criar e otimizar os processos de trabalho para os desenvolvedores de software, com isso, descartam-se as atividades de levantamento de requisitos, modelagem, design e codificação, pois não são consideradas típicas de um engenheiro de software, embora em muitas circunstâncias, é ele que executa tais atividades, e ao mesmo tempo, tem o foco em observar, avaliar, orientar e alterar os processos produtivos quando for preciso (WAZLAWICK, 2013).

A engenharia de software tem duas definições, uma delas é que a engenharia pode fazer qualquer produto funcionar, aplicando teorias, métodos e ferramentas quando for necessário, sempre procurando soluções para os problemas. Ela não se dedica apenas aos processos técnicos do desenvolvimento de software, mas também ao gerenciamento de projetos e o desenvolvimento de ferramentas, métodos e teorias para apoiar a produção do software. Os engenheiros de software se preocupam mais com os serviços, as interações e as restrições dentro das quais o sistema deve ser construído e operado (SOMMERVILLE, 2007)

As exigências para o desenvolvimento de um software são cada dia mais relevantes, pois empresas, órgãos e organizações precisam cada vez mais do mesmo para se manter em crescimento no mercado, isso conta com a enorme quantidade de informações indispensáveis para obter a competitividade, e tomar decisões prudentes no seu exato momento. Portanto, é preciso obter modificações do software para que ele possa moldar-se de acordo com o cenário proposto, ou seja, o software deve ser sujeito a qualquer tipo de manutenção (PRESSMAN, 2011).

Segundo WAZLAWICK (2013) a engenharia de software apresenta um conjunto de princípios que são usados no desenvolvimento de projetos, os quais são apresentados a seguir:

- a) Decomposição: construção de softwares complexos a partir de partes cada vez mais simples.
- b) Abstração: executar um elemento em uma linguagem de nível mais alto do que for preciso para sua construção.
- c) Generalização: é usada na classificação de dados. Ou seja, objetos podem ser classificados em várias classes simultaneamente.
- d) Padronização: elaboração de padrões de programações ajuda no desenvolvimento de produtos com uma qualidade significativa.
- e) Flexibilidade: facilidade para acomodar as mudanças nos requisitos que não foram previstas.
- f) Formalidade: desenvolver um produto com tal formalidade que tenha que ser atingida.
- g) Rastreabilidade: produzir especificações para que saiba quando ocorre alguma alteração.
- h) Desenvolvimento iterativo: realizar vários ciclos distintos para o desenvolvimento de um processo para o amadurecimento do produto final.
- i) Gerenciamento de Requisitos: sabendo que os requisitos mudam frequentemente, é necessário gerenciar todas essas mudanças para o desenvolvimento e evolução do software.

j) Arquiteturas baseadas em componentes: melhor forma de se adquirir reusabilidade e para saber lidar com a complexidade.

k) Modelagem visual: é possível produzir sistemas completos sem passar nas fases mais clássicas da programação.

l) Verificação contínua da qualidade: a qualidade tem que obter total atenção do começo ao fim de qualquer projeto de desenvolvimento.

m) Controle de mudanças: permite que volte atrás em um caminho de desenvolvimento ou até produzir em paralelo de diferentes versões do mesmo componente.

n) Gerenciamento de riscos: realiza atividades preventivas para riscos ou até mesmo a probabilidade do mesmo.

4.6.1 TESTE DE SOFTWARE

Antes de iniciar a discussão sobre teste de software é preciso esclarecer alguns conceitos relacionados a essa atividade. Inicialmente, faz-se necessário conhecer a diferença entre Defeitos, Erros e Falhas. As definições que serão usadas aqui seguem a terminologia padrão para Engenharia de Software do IEEE – Institute of Electrical and Electronics Engineers – (IEEE) de 1990 (DIAS NETO, 2010).

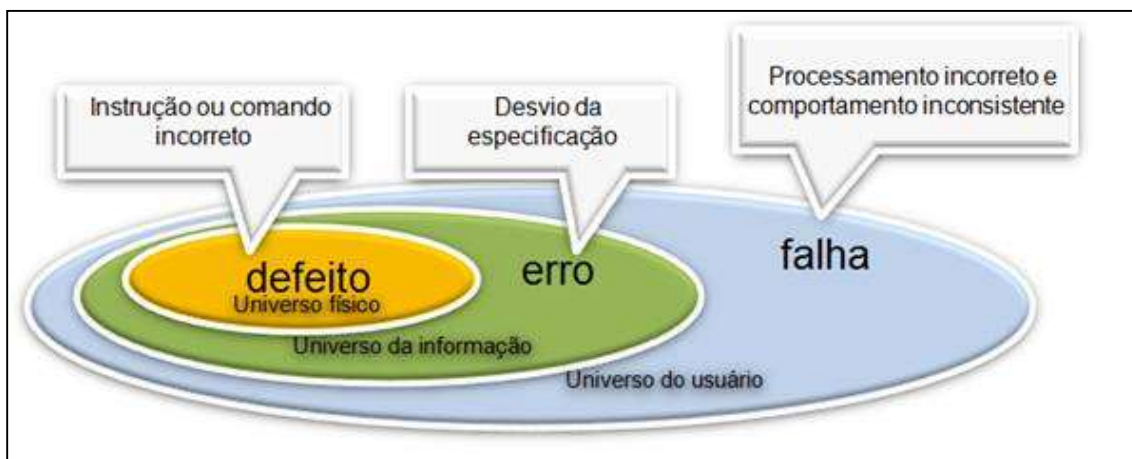
a) Defeito é um ato inconsistente cometido por um indivíduo ao tentar entender uma determinada informação, resolver um problema ou utilizar um método ou uma ferramenta. Por exemplo, uma instrução ou comando incorreto.

b) Erro é uma manifestação concreta de um defeito num artefato de software. Diferença entre o valor obtido e o valor esperado, ou seja, qualquer estado intermediário incorreto ou resultado inesperado na execução de um programa constitui um erro.

c) Falha é o comportamento operacional do software diferente do esperado pelo usuário. Uma falha pode ter sido causada por diversos erros e alguns erros podem nunca causar uma falha.

A Figura 1 expressa a diferença entre esses conceitos.

Figura 1 – Defeito x erro x falha.



Fonte: Dias Neto (2010).

Como pode ser observado na Figura 1, defeitos fazem parte do universo físico (a aplicação propriamente dita) e são causados por pessoas, por exemplo, através do mal uso de uma tecnologia, podendo ocasionar a manifestação de erros em um produto, ou seja, a construção de um software de forma diferente ao que foi especificado (universo de informação). Por fim, os erros geram falhas, que são comportamentos inesperados em um software, que afetam diretamente o usuário final da aplicação (universo do usuário), e podem inviabilizar a utilização de um software.

Teste de software pode ser definido como sendo o processo de execução de um produto para determinar se ele atingiu suas especificações e funcionou corretamente no ambiente para o qual foi projetado, ao mesmo tempo em que faz parte de todo o processo de engenharia de software. O seu objetivo é revelar falhas em um produto, para que as causas dessas falhas sejam identificadas e possam ser corrigidas pela equipe de desenvolvimento antes da entrega final. Por conta dessa característica das atividades de teste, diz-se que sua natureza é “destrutiva”, e não “construtiva”, pois visa ao aumento da confiança de um produto através da exposição de seus problemas, porém antes de sua entrega ao usuário final (DIAS NETO, 2010).

Ainda segundo o autor, o conceito de teste de software pode ser compreendido através de uma visão intuitiva ou mesmo de uma maneira formal. Existem atualmente várias definições para esse conceito. De uma forma simples, testar um software significa verificar através de uma execução controlada se o seu comportamento ocorre de acordo com o especificado. O objetivo principal desta tarefa é revelar o número máximo de falhas dispondo do mínimo de esforço, ou seja, mostrar aos que desenvolvem se os resultados estão ou não de acordo com os padrões estabelecidos (DIAS NETO, 2010).

Para Pressman (2011), o processo de testes visa encontrar falhas no sistema para corrigi-las antes de distribuir o sistema, ou de atualizar o software com novos recursos, portanto o sistema deve ser projetado e implementado pensando na facilidade da realização dos testes. Por sua vez os testes devem ser feitos levando em consideração certas características para que nada seja deixado para trás.

Em conformidade com o autor, o conceito de testabilidade consiste em se medir o quão simples é o ato de testar um software. As características apresentadas a seguir caracterizam um software a ser testável:

- a) Operabilidade: um sistema projetado e implementado tendo em mente a qualidade, terá poucas falhas quando os testes forem realizados.
- b) Observabilidade: quando é possível ver com clareza as entradas, saídas e variáveis do sistema fica mais fácil de detectar possíveis falhas.
- c) Controlabilidade: entradas geram saídas específicas, e para cada tipo de saída existirá um tipo de entrada específica. Se o engenheiro puder controlar essas entradas ficará mais fácil realizar os testes.
- d) Decomponibilidade: o sistema é construído a partir de módulos e pode ser testado em partes.
- e) Simplicidade: Quanto mais simples um sistema for, atingindo o objetivo, mais simples serão os testes.
- f) Estabilidade: Quanto menos alterações o software tiver, menos testes precisarão ser feitos.

g) Compreensibilidade: Quanto mais informações estiverem disponíveis para o entendimento do software, mais eficazes serão os testes, isso inclui manuais organizados, detalhados e especificados.

A atividade de teste é composta por alguns elementos essenciais que auxiliam na formalização desta atividade, os quais serão apresentados a seguir.

a) Caso de Teste: descreve uma condição particular a ser testada e é composto por valores de entrada, restrições para a sua execução e um resultado ou comportamento esperado (CRAIG; JASKIEL, 2002).

b) Procedimento de Teste: é uma descrição dos passos necessários para executar um caso (ou um grupo de casos) de teste (CRAIG; JASKIEL, 2002).

c) Critério de Teste: serve para selecionar e avaliar casos de teste de forma a aumentar as possibilidades de provocar falhas ou, quando isso não ocorre, estabelecer um nível elevado de confiança na correção do produto (ROCHA et al., 2001). Os critérios de teste podem ser utilizados como:

i. Critério de Cobertura dos Testes: permite a identificação de partes do programa que devem ser executadas para garantir a qualidade do software e indicar quando o mesmo foi suficientemente testado (RAPPS; WEYUKER, 1982). Ou seja, determinar o percentual de elementos necessários por um critério de teste que foram executados pelo conjunto de casos de teste.

ii. Critério de Adequação de Casos de Teste: Quando, a partir de um conjunto de casos de teste T qualquer, ele é utilizado para verificar se T satisfaz os requisitos de teste estabelecidos pelo critério. Ou seja, este critério avalia se os casos de teste definidos são suficientes ou não para avaliação de um produto ou uma função (ROCHA et al., 2001).

iii. Critério de Geração de Casos de Teste: quando o critério é utilizado para gerar um conjunto de casos de teste T adequado para um produto ou função, ou seja, este critério define as regras e diretrizes para geração dos casos de teste de um produto que esteja de acordo com o critério de adequação definido anteriormente (ROCHA et al., 2001).

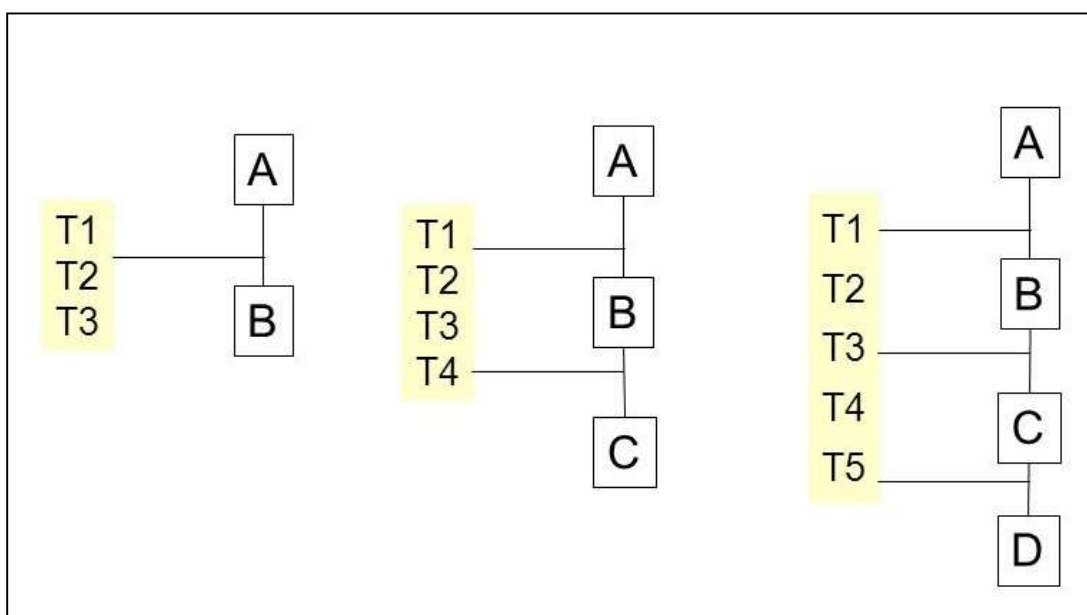
4.5.2 NÍVEIS DE TESTE DE SOFTWARE

O planejamento dos testes deve ocorrer em diferentes níveis e em paralelo ao desenvolvimento do software (Figura 4). Segundo Rocha et al. (2001) os principais níveis de teste de software são:

Teste de Unidade: também conhecido como teste unitário. Tem por objetivo explorar a menor unidade do projeto, procurando provocar falhas ocasionadas por defeitos de lógica e de implementação em cada módulo, separadamente. O universo alvo desse tipo de teste são os métodos dos objetos ou mesmo pequenos trechos de código.

Teste de Integração: O processo de integração do sistema engloba a criação de um sistema com base em seus componentes e o teste do sistema derivado da interação de diferentes componentes. Os componentes podem ser comerciais, reusáveis adequados a certo sistema ou novos componentes desenvolvidos. Neste teste é verificado se todos os componentes são chamados corretamente, se transmitem os dados corretamente e no seu devido tempo e se funcionam “como um relógio”, ou seja, trabalhando em conjunto e ou um dependendo do outro (SOMMERVILLE, Figura 2, pode-se entender melhor o funcionamento do teste de integração).

Figura 2 - Testes de integração incremental.

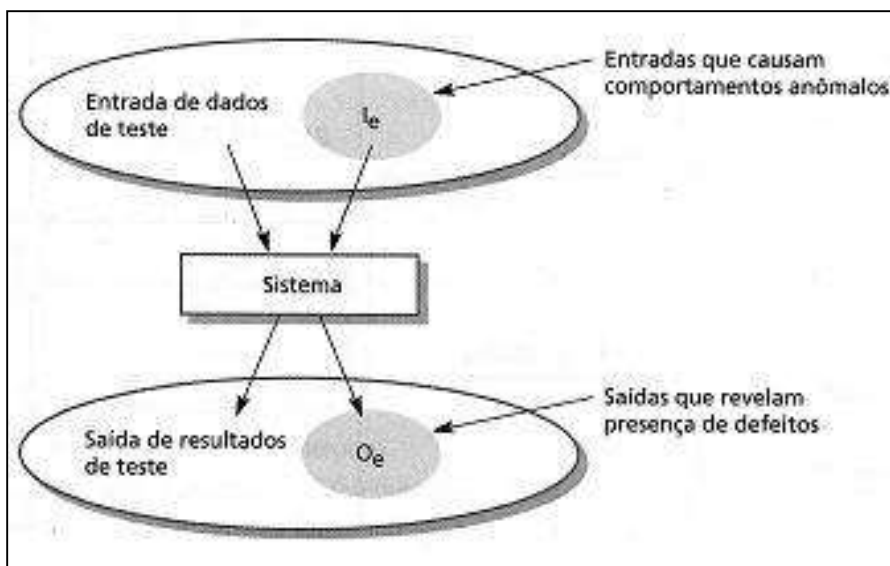


Fonte: Elaborada pelo autor.

“[...] A, B, C e D são componentes e T1 a T5 referem – se aos testes de características incorporadas ao sistema. T1, T2 e T3 são executados inicialmente no sistema composto dos componentes A e B (sistema mínimo). Se esses testes revelarem defeitos, estes serão corrigidos. O componente C é integrado e T1, T2 e T3 são repetidos para assegurar que não há interação inesperadas entre A e B. Se surgirem problemas nesses testes, isso provavelmente significa que eles são decorrentes da interação com o novo componente. A origem do problema é localizada, simplificando a identificação e correção do defeito. O conjunto de testes T4 é também executado no sistema. Finalmente, o componente D é integrado e testado por meio dos testes existentes e dos novos testes (T5)” (SOMMERVILLE, 2007, p. 358).

Teste de Realeases - também chamado como teste caixa-preta e teste funcional, nessa etapa é testado o release do sistema que será entregue ao usuário. Tem como objetivo aumentar a confiança do desenvolvedor, no qual saberá se o sistema atende a todos os requisitos, ou seja, se ele atender a todos os requisitos ele poderá ser entregue ao cliente ou ao seu destino final. Mas para isso o sistema deve fornecer funcionalidade, desempenho, confiabilidade e que não forneça nenhum tipo de falhas durante sua execução (SOMMERVILLE, 2007). Ainda segundo o autor, no teste de realeases, a equipe de teste mune entradas no sistema para verificar se as saídas serão ou não as esperadas, se a resposta for não, algum problema foi detectado. Como especificado na Figura 3.

Figura 3 – Teste de caixa preta.



Fonte: Sommerville (2007, p.359).

Pode-se afirmar que na área de teste de software, o teste em si não comprova que o produto está desprovido de defeitos (WAZLAWICK, 1967).

Teste de Sistema - avalia o software em busca de falhas por meio da utilização do mesmo, como se fosse um usuário final. Dessa maneira, os testes são executados nos mesmos ambientes, com as mesmas condições e com os mesmos dados de entrada que um usuário utilizaria no seu dia-a-dia de manipulação do software. Verifica se o produto satisfaz seus requisitos.

Teste de Aceitação - são realizados geralmente por um restrito grupo de usuários finais do sistema. Esses simulam operações de rotina do sistema de modo a verificar se seu comportamento está de acordo com o solicitado.

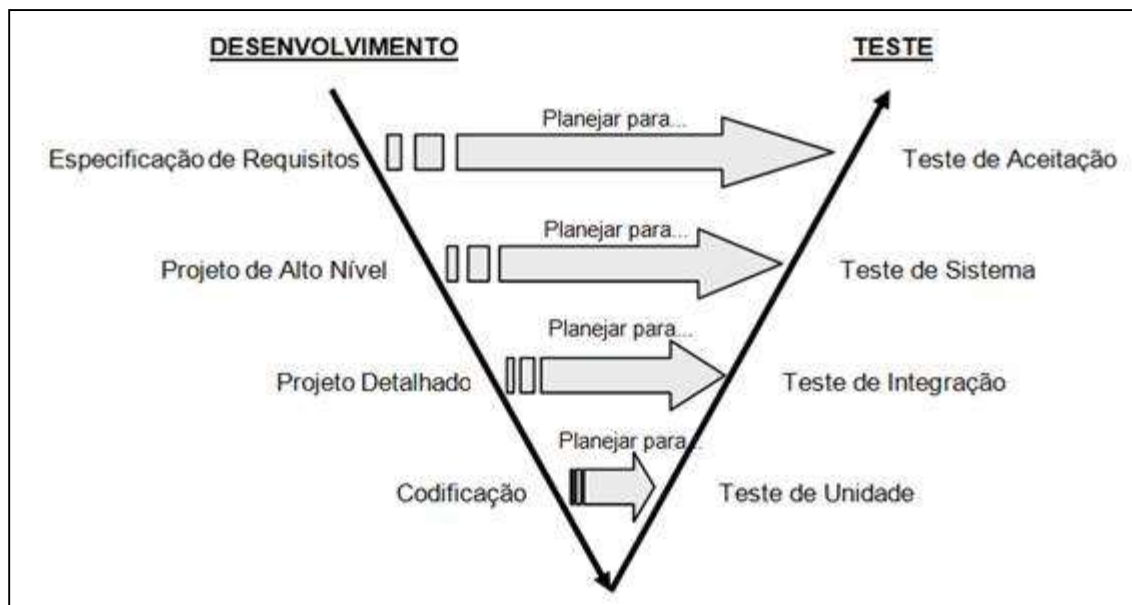
Teste de Regressão - não corresponde a um nível de teste, mas é uma estratégia importante para redução de “efeitos colaterais”. Consiste em se aplicar, a cada nova versão do software ou a cada ciclo, todos os testes que já foram aplicados nas versões ou ciclos de teste anteriores do sistema. Pode ser aplicado em qualquer nível de teste.

Dessa forma, conforme traz a Figura 13, o planejamento e projeto dos testes devem ocorrer de cima para baixo, ou seja:

Inicialmente é planejado o teste de aceitação a partir do documento de requisitos. Após isso é planejado o teste de sistema a partir do projeto de alto nível do software. Em seguida ocorre o planejamento dos testes de integração

a partir o projeto detalhado. E por fim, o planejamento dos testes a partir da codificação.

Figura 13 - Modelo V descrevendo o paralelismo entre as atividades de desenvolvimento e teste de software.



Fonte: Craig e Jaskiel (2002).

4.5.3 TÉCNICAS DE TESTE DE SOFTWARE

Atualmente existem muitas maneiras de se testar um software. Mesmo assim, existem as técnicas que sempre foram muito utilizadas em sistemas desenvolvidos sobre linguagens estruturadas que ainda hoje têm grande valia para os sistemas orientados a objeto. Apesar dos paradigmas de desenvolvimento serem diferentes, o objetivo principal destas técnicas continua a ser o mesmo: encontrar falhas no software.

De acordo com Dias Neto (2010) as técnicas de teste são classificadas de acordo com a origem das informações utilizadas para estabelecer os requisitos de teste. Elas contemplam diferentes perspectivas do software, e impõe-se a necessidade de se estabelecer uma estratégia de teste que contemple as vantagens e os aspectos complementares dessas técnicas. As técnicas existentes são: técnica funcional e estrutural.

Outras técnicas de teste que podem e devem ser utilizadas de acordo com necessidades de negócio ou restrições tecnológicas são: teste de desempenho, teste de usabilidade, teste de carga, teste de stress, teste de confiabilidade e teste de recuperação.

Segundo Dias Neto (2010) alguns autores chegam a definir uma técnica de teste caixa cinza, que seria um mesclado do uso das técnicas de caixa preta e caixa branca, mas, como toda execução de trabalho relacionado à atividade de teste utilizará simultaneamente mais de uma técnica de teste, é recomendável que se fixe os conceitos primários de cada técnica.

5 METODOLOGIA

O desenvolvimento foi dividido em quatro etapas, cada etapa contou com uma seção teórica e uma prática (implementação dos conteúdos no software).

5.1 LEVANTAMENTO BIBLIOGRÁFICO

A parte inicial teórica deste trabalho envolveu uma pesquisa exploratória com o objetivo de desenvolver o tema e trabalhar para melhor desenvolver uma solução plausível e acessível. Não só sendo de extrema importância para contextualização, mas também para um maior entendimento do problema.

Portanto, um levantamento bibliográfico foi o primeiro passo para a construção da pesquisa, avaliando bibliografias em livros, artigos científicos, buscando um aprofundamento maior relacionado ao desenvolvimento de software voltado, sobretudo ao público-alvo final desta investigação: crianças com espectro autista. Cabe destacar que todas as informações referentes ao público-alvo foram realizadas por meio de levantamento bibliográfico e não diretamente com as crianças.

5.2 LINGUAGEM DE PROGRAMAÇÃO

Nesta etapa deste projeto, foi realizado um estudo acerca dos comandos e possibilidades de linguagens de programação candidatas que poderão ser utilizadas como ferramentas para o desenvolvimento do software.

Nesta etapa também foram levantados possíveis ambientes (frameworks) para a programação. Aspectos como facilidade de programação, custos, desempenho, dentre outros, serão considerados na escolha.

5.3 DEFINIÇÃO DAS ATIVIDADES

Primeiramente, foi feita uma análise de formas pedagógicas de ensinar matemática para crianças do ensino fundamental I (operações básicas: adição,

subtração, multiplicação e divisão). Este levantamento foi feito tendo por base bibliografias científicas a respeito desta temática.

Este processo contou com a elaboração de atividades que foram adaptadas e introduzidas no jogo eletrônico, além disso, esses exercícios são correspondentes às crianças autistas com dificuldade em matemática.

5.4 JOGOS DIGITAIS

Nesta etapa, foram estudados alguns conceitos importantes que norteiam o desenvolvimento de jogos digitais. Além disso, é nessa fase onde foram estudadas formas de adaptar as atividades matemáticas para jogos eletrônicos.

5.5 INTERFACE

É indiscutível a importância da interface para o software como um todo, e uma interface atrativa torna-se indispensável em softwares voltados para crianças. Nessa fase, foi desenvolvida e implementada toda a interface (*front-end*).

Para esse processo, foram necessárias as seguintes etapas:

- Cores: estudo sobre conceitos de design. (como as cores podem ser dispostas e quais são mais atrativas para crianças autistas).
- Sons: como os sons influenciam na concentração das crianças.
- Personagem: foi desenvolvido um personagem que tem como função se comunicar com a criança através de
- mensagens explicativas, positivas e de incentivo – já que crianças autistas aprendem de forma direta.

Cada criança com autismo se comporta de forma distinta diante de determinadas situações – considerando isso, cada um dos itens acima foi elaborado de forma que o jogo se tornasse atrativo para todas as crianças independente de suas dificuldades.

6. RESULTADOS

Os resultados do projeto são apresentados a seguir.

6.1 TECNOLOGIAS E FERRAMENTAS UTILIZADAS

Para trabalhar com as tecnologias foram usados o Visual Studio Code e o Visual Studio 2019 (ambos Microsoft). Estas tecnologias estão divididas em duas vertentes: *Front-end* e *Back-end*.

No *Front-end (client side)* foi utilizado o framework Angular (google) utilizando *HTML*, *CSS* e *Typescript* (linguagem baseada em *javascript*, porém as variáveis podem ser tipadas). Já no *Back-end (server side)* foi utilizado .Net Core na versão 3.1.8, onde foi possível trabalhar toda a estrutura do projeto. Unido ao *Back-end* foi utilizado o *Entity Framework*, uma ferramenta de persistência que facilita a interação com o banco de dados e utiliza uma abordagem chamada *Code First*, o que significa que todas as alterações estruturais que serão feitas na base são geradas via código e aplicadas a mesma através do *migration*.

Para construção dos itens da interface e de toda a identidade visual do jogo foi utilizado o software Adobe Illustrator CC 2019, onde foi possível aplicar a técnica de design conhecido como *Flat Design*. Segundo a gráfica Forma Certa (2018), “O flat design é uma tendência do web design que é a pura simplicidade dos elementos, da clareza do layout. Se distingue por suas formas limpas e planas. O conceito funciona sem variações na estrutura do layout, como sombras, chanfros, relevo, gradientes ou outras ferramentas que adicionam profundidade. Cada elemento ou caixa é nítido e não tem arestas ou sombras. Nada realista é adicionado.”. A simplicidade e clareza dos elementos facilitam o entendimento, deixando o jogo muito mais intuitivo para as crianças.

6.2 FUNCIONAMENTO DO JOGO

Primeiramente, o jogador deverá preencher seu nome como mostra a Figura 14:

Figura 14 – Tela inicial do jogo.



Fonte: Elaborado pela Autora.

Após registrar seu nome, o jogador terá acesso ao menu do jogo (Figura 15).

Figura 15 – Menu do jogo.



Fonte: Elaborado pela Autora.

Ao clicar em “Jogar”, o usuário será redirecionado para a primeira fase do jogo. E conforme for acertando os resultados será redirecionado para as fases seguintes, como ilustrado nas Figuras (16 à 30) a seguir:

Figura 16 – Fase um.



Fonte: Elaborado pela Autora.

Figura 17 – Fase dois.



Fonte: Elaborado pela Autora.

Figura 18 – Fase três.



Fonte: Elaborado pela Autora.

Figura 19 – Fase quatro.



Fonte: Elaborado pela Autora.

Figura 20 – Fase cinco.



Fonte: Elaborado pela Autora.

Figura 21 – Fase seis.



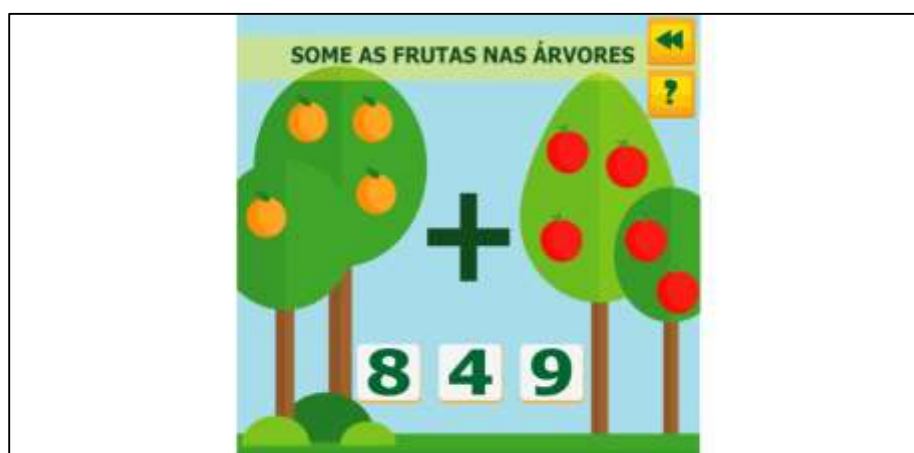
Fonte: Elaborado pela Autora.

Figura 22 – Fase sete.



Fonte: Elaborado pela Autora.

Figura 23 – Fase oito.



Fonte: Elaborado pela Autora.

Figura 24 – Fase nove.



Fonte: Elaborado pela Autora.

Figura 25 – Fase dez.



Fonte: Elaborado pela Autora.

Figura 26 – Fase onze.



Fonte: Elaborado pela Autora.

Figura 27 – Fase doze.



Fonte: Elaborado pela Autora.

Figura 28 – Fase treze.



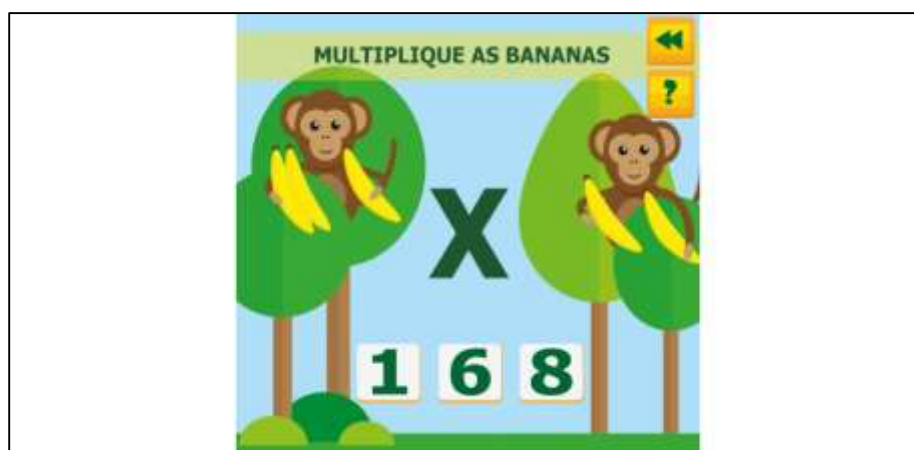
Fonte: Elaborado pela Autora.

Figura 29 – Fase catorze.



Fonte: Elaborado pela Autora.

Figura 30 – Fase quinze.



Fonte: Elaborado pela Autora.

Como é possível observar nas figuras anteriores, todas as fases possuem dois botões superiores a direita. Esses botões abrigam as funções “Atalho” e “Dica”, como é possível observar na Figura 31:

Figura 31 – Atalhos.



Atalhos. Fonte: Elaborado pela Autora.

Figura 32 – Dica.



Dica. Fonte: Elaborado pela Autora.

No botão “Atalhos” é possível observar as funções aumentar volume, abaixar volume e pausar música, além disso, também é possível ser redirecionado ao menu através do botão “Voltar ao menu”. Já ao clicar no botão “Dica” (Figura 32), o jogador poderá ter acesso a uma ajuda referente a fase em que está. Na figura 32, podemos observar a dica pertencente a fase quatro.

Em cada fase, o usuário irá receber uma mensagem após responder a questão. Caso a resposta esteja correta a mensagem “Você acertou! Parabéns.” (Figura 34) será exibida em tela, e caso esteja incorreta o mesmo será instigado a tentar novamente através da mensagem “Vamos tentar novamente?” (Figura 33).

Figura 33 – Mensagem erro.



Fonte: Elaborado pela Autora.

Figura 34 – Mensagem acerto.



Fonte: Elaborado pela Autora.

7 CONSIDERAÇÕES FINAIS

Como trabalhos futuros para o projeto sugere-se inseri-lo de fato no cotidiano das crianças, como forma de apoio ao ensino, já que esta etapa de teste junto ao usuário final não estava prevista nesta investigação. Os resultados do emprego do jogo no dia a dia das crianças são importantes para uma melhor avaliação dos estudos realizados e metodologias propostas neste projeto. Dessa forma, será possível analisar os efeitos da disposição dos conteúdos e interface do jogo no aprendizado e compreensão da matemática básica para crianças autistas.

Outra tarefa futura para este projeto é a possibilidade de construir uma aplicação mobile, explorando outras tecnologias, linguagens de programação e até mesmo *engines* para desenvolvimento de jogos (como o Unity). Dessa forma, o jogo se tornará ainda mais acessível, e poderá ser utilizado de forma mais abrangente.

É inegável o tamanho aprendizado que este projeto trouxe. Conhecimentos importante como: desenvolvimento web, *frameworks*, ferramentas de persistência (banco de dados), *back-end*, entre outros aprendizados que ostentaram tamanha experiência, auxiliando no desenvolvimento profissional e social (considerando a importância de uma educação inclusiva e acessível a todos).

8 REFERÊNCIAS BIBLIOGRÁFICAS

Abrantes. Matemática Básica: qual a importância?. Stoodi, 2017. Disponível em: <<https://www.stoodi.com.br/blog/2017/04/06/3-motivos-para-voce-incluir-matematica-basica-no-seu-plano-de-estudos/>>. Acesso em: 24 fev. 2020.

A ciência e a tecnologia como estratégia de desenvolvimento. Centro de pesquisa em Ciência, Tecnologia e Sociedade (IPEA), 2019. Disponível em: <<http://www.ipea.gov.br/cts/pt/central-de-conteudo/artigos/artigos/116-a-ciencia-e-a-tecnologia-como-estrategia-de-desenvolvimento>>. Acesso em: 10 fev. 2020.

Berehoffa, A. M. Designer cria modelo de jogo virtual para crianças autistas. Universidade de Brasília (UnB CIÊNCIA), 2009. Disponível em: <<https://www.unbciencia.unb.br/artes-e-letras/105-desenho-industrial/325-designer-cria-modelo-de-jogo-virtual-para-criancas-autistas>>. Acesso em: 25 fev. 2020.

Brites, C. Métodos de Alfabetização para Crianças Autistas. Entendendo Autismo, 2020. Disponível em: <<http://entendendoautismo.com.br/artigo/alfabetizacao-para-criancas-autistas/>>. Acesso em: 8 fev. 2020.

Brites, C. Como ensinar matemática para crianças com autismo?. Entendendo Autismo, 2020. Disponível em: <<http://entendendoautismo.com.br/artigo/como-ensinar-matematica-para-criancas-com-autismo/>>. Acesso em: 8 fev. 2020.

Brites, C. ABA: Conheça essa abordagem para crianças com autismo. Entendendo Autismo, 2020. Disponível em: <<http://entendendoautismo.com.br/artigo/aba-conheca-essa-abordagem-para-criancas-com-autismo/>>. Acesso em: 8 fev. 2020.

Brites, C. Características cognitivas do autismo. Entendendo Autismo, 2020. Disponível em: <<http://entendendoautismo.com.br/artigo/caracteristicas-cognitivas-autismo/>>. Acesso em: 8 fev. 2020.

PRESSMAN, Roger. Engenharia de Software: uma abordagem profissional. 7 ed. São Paulo: McGraw-Hill, 2011.

Brites, C. Conheça as características do Autismo Infantil. NeuroSaber, [2019?]. Disponível em: <<https://neurosaber.com.br/conheca-as-caracteristicas-do-autismo-infantil/>>. Acesso em: 3 mar. 2020

Brito, G. da S.; Novôa, J. Transtorno do espectro autista: As tecnologias como ferramentas de ensino na educação especial. XIII Congresso Nacional de Educação (EDUCERE), [2017?]. Disponível em: <https://educere.bruc.com.br/arquivo/pdf2017/23477_12977.pdf>. Acesso em: 8 fev. 2020.

CRAIG, Rick D.; JASKIEL, Stefan P. Systematic Software Testing. Norwood: Artech House Publishers, 2002.

Detlinger, J. Brincadeira é coisa séria: saiba a importância das cores e formas no desenvolvimento do seu filho. Pais e Filhos – UOL, 2018. Disponível em: <<https://paisefilhos.uol.com.br/crianca/brincadeira-e-coisa-seria-saiba-a-importancia-das-cores-e-formas-no-desenvolvimento-do-seu-filho/>>. Acesso em: 25 fev. 2020.

DIAS NETO, Arilo Cláudio. Introdução a Teste de Software. Revista Engenharia de Software Magazine, n. 1, p. 54-9, jun.2010.

España, D. Tecnologia e Humanidade. Exame, 2017. Disponível em: <<https://exame.abril.com.br/blog/o-que-te-motiva/tecnologia-e-humanidade/>>. Acesso em: 10 fev. 2020.

Fabossi, L. A. Design de aplicativo para tablets para crianças em educação infantil. Universidade Tecnológica Federal do Paraná, 2014. Disponível em: <http://repositorio.roca.utfpr.edu.br/jspui/bitstream/1/6847/1/CT_CODEG_2013_2_14.pdf.pdf>. Acesso em: 24 fev. 2020.

Mont'Alvão, C.; Carus, Ai. Design de interfaces em software educacionais: algumas recomendações a partir da aplicação do método co-descoberta. Pontifícia Universidade Católica do Rio de Janeiro, [2012]. Disponível em: <<https://www.maxwell.vrac.puc-rio.br/20809/20809.PDF>>. Acesso em: 24 fev. 2020.

Jorge, A. M.; Basniak M. I. O ensino das quatro operações básicas da matemática com o uso do ensino exploratório no 6º ano do ensino fundamental. Web Artigos, 2014. Disponível em: <
http://www.diaadiaeducacao.pr.gov.br/portals/cadernospde/pdebusca/producoes_pde/2016/2016_artigo_mat_unespar-uniaodavitoria_andreiamourajorge.pdf>. Acesso em: 24 fev. 2020.

Kashiwakura, E. Y. Jogando e Aprendendo: um paralelo entre videogames e habilidades cognitivas. Pontifícia Universidade Católica de São Paulo (PUC-SP), 2008. Disponível em: <
<https://tede2.pucsp.br/bitstream/handle/18215/1/Eduardo%20Yukio%20Kashiwakura.pdf>>. Acesso em: 10 fev. 2020.

Kohn, K.; Moraes, C. H. de. O impacto das novas tecnologias na sociedade: conceitos e características da Sociedade da Informação e da Sociedade Digital. Sociedade Brasileira de Estudos Interdisciplinares da Comunicação (Intercom), 2007. Disponível em: <
<https://www.intercom.org.br/papers/nacionais/2007/resumos/R1533-1.pdf>>. Acesso em: 10 fev. 2020.

Lara, I. C. de. O jogo como estratégia de ensino de 5ª a 8ª série. VIII Encontro Nacional de educação matemática, 2004. Disponível em: <
<http://www.sbemrasil.org.br/files/viii/pdf/02/MC63912198004.pdf>>. Acesso em: 22 fev. 2020.

Lara, I. C. de. Jogando com a matemática de 5ª a 8ª série. [2003]. Disponível em: <
http://miltonborba.org/CD/Interdisciplinaridade/Encontro_Gaicho_Ed_Matem/minicursos/MC53.pdf>. Acesso em: 13 fev. 2020.

Marketing Lecom. Descubra os Principais Impactos da Tecnologia na Sociedade Atual. BlogLecom, 2019. Disponível em: <
<https://www.lecom.com.br/blog/descubra-os-principais-impactos-da-tecnologia-na-sociedade-atual/>>. Acesso em: 10 fev. 2020.

Nunes, J. V.; Gonçalves, B. S. Hipermídia para aprendizagem no contexto da EAD: uma avaliação a partir de métodos prospectivos. Estudos em Design - Revista (online), [2014]. Disponível em: <<https://estudosemdesign.emnuvens.com.br/design/article/view/97/94>>. Acesso em: 25 fev. 2020.

Olivério, J. B. O ensino da matemática através do lúdico na educação infantil. Brasil Escola, [200-]. Disponível em: <<https://meuartigo.brasilecola.uol.com.br/pedagogia/o-ensino-matematica-atraves-ludico-na-educacao-infantil.htm>>. Acesso em: 24 fev. 2020.

Organização Pan-americana da saúde (OPAS). Folha informativa - Transtorno do espectro autista. 2017. Disponível em: <<https://www.paho.org/bra/index.php?Itemid=1098>>. Acesso em: 4 mar. 2020.

Prudêncio, M. A tecnologia do dia-a-dia. Correio Braziliense, 2012. Disponível em: <https://www.correiobraziliense.com.br/app/noticia/tecnologia/2018/03/13/interna_tecnologia,665761/a-tecnologia-do-dia-a-dia.shtml>. Acesso em: 10 fev. 2020.

Re, A. Del. Tecnologia é forte aliada para o desenvolvimento de pessoas com autismo. Estadão, 2018. Disponível em: <<https://emails.estadao.com.br/blogs/familia-plural/tecnologia-e-forte-aliado-para-o-desenvolvimento-de-pessoas-com-autismo/>>. Acesso em: 8 fev. 2020.

RAPPS, S.; WEYUKER, E. J. *Data Flow Analysis Techniques for Program Test Data Selection*. In: 6th International Conference on Software Engineering, Tokio, Japan, 1982, p. 272–78.

Rede brasileira de cidades inteligentes e humanas. 10 Vantagens que os jogos educativos trazem para melhorar o desempenho dos alunos. [201-]. Disponível em: <procurar link>. Acesso em: 10 fev. 2020.

ROCHA, Ana Regina Cavalcanti da, MALDONADO, José Carlos; WEBER, Kival Cjaves (Orgs.). Qualidade de Software: teoria e prática. São Paulo: Prentice Hall, 2001.

Rodrigues, Evangelista M. Et al. **A matemática na educação infantil**. Revista Científica Multidisciplinar Núcleo do Conhecimento, 2003. Disponível em: <<https://www.nucleodoconhecimento.com.br/pedagogia/matematica-na-educacao>>. Acesso em: 24 fev. 2020.

Prudêncio, M. A tecnologia do dia-a-dia. Correio Braziliense, 2012. Disponível em:

<https://www.correiobraziliense.com.br/app/noticia/tecnologia/2018/03/13/interna_tecnologia,665761/a-tecnologia-do-dia-a-dia.shtml>. Acesso em: 10 fev. 2020.

Re, A. Del. Tecnologia é forte aliada para o desenvolvimento de pessoas com autismo. Estadão, 2018. Disponível em:

<<https://emails.estadao.com.br/blogs/familia-plural/tecnologia-e-forte-aliado-para-o-desenvolvimento-de-pessoas-com-autismo/>>. Acesso em: 8 fev. 2020.

RAPPS, S.; WEYUKER, E. J. *Data Flow Analysis Techniques for Program Test Data Selection*. In: 6th International Conference on Software Engineering, Tokio, Japan, 1982, p. 272–78.

Rede brasileira de cidades inteligentes e humanas. 10 Vantagens que os jogos educativos trazem para melhorar o desempenho dos alunos. [201-]. Acesso em: 10 fev. 2020.

ROCHA, Ana Regina Cavalcanti da, MALDONADO, José Carlos; WEBER, Kival Cjaves (Orgs.). Qualidade de Software: teoria e prática. São Paulo: Prentice Hall, 2001.

Savi, R.; Ulbricht, V. R. Jogos digitais educacionais: benefícios e desafios. CINTED-UFRGS, 2008. Disponível em: <<https://www.seer.ufrgs.br/renote/article/viewFile/14405/83100>>. Acesso em: 10 fev. 2020.

Silva, M. F. da. A importância da matemática no ensino fundamental. Revista Faculdade Eficaz, [2014?]. Disponível em: <https://revista.faculdadeeficaz.edu.br/artigos/SILVA_Michele%20FI_22-07-2015.pdf>. Acesso em: 24 fev. 2020.

Silveira R. M. C. F; Bazzo W. A. CIÊNCIA E TECNOLOGIA: Transformando a relação do ser humano com o mundo. IX Simpósio Internacional Processo Civilizador, [200-]. Disponível em: <<http://www.uel.br/grupo-estudo/processoscivilizadores/portugues/sites/anais/anais9/artigos/workshop/art19.pdf>>. Acesso em: 10 fev. 2020.

Super Interessante. Softwares Educacionais: aventuras em classe. 2016. Disponível em: <<https://super.abril.com.br/tecnologia/softwares-educacionais-aventuras-em-classe/>>. Acesso em: 22 fev. 2020.

SOMMERVILLE, Ian. Engenharia de Software. 8 ed. São Paulo: Pearson AddisonWesley, 2007.

Szafran, V. Robô ensina matemática e habilidades sociais a crianças com autismo. Olhar Digital, 2020. Disponível em: <<https://olhardigital.com.br/noticia/robo-ensina-matematica-e-habilidades-sociais-a-criancas-com-autismo/97367>>. Acesso em: 4 mar. 2020.

Tenorio, G.; Pinheiro, C. O que é autismo, das causas aos sinais e o tratamento. Saúde – Abril, 2018. Disponível em: <<https://saude.abril.com.br/mente-saudavel/o-que-e-autismo-das-causas-aos-sinais-e-o-tratamento/>>. Acesso em: 4 mar. 2020.

Varella, D. Transtorno do Espectro Autista (TEA). Drauzio Varella – UOL, [201-]. Disponível em: <<https://drauziovarella.uol.com.br/doencas-e-sintomas/transtorno-do-espectro-autista-tea/>>. Acesso em: 3 mar. 2020.

Vieira, F. Importância dos recursos tecnológicos na sociedade contemporânea. Portal de e-governo, inclusão digital e sociedade do conhecimento, 2012. Disponível em: <<http://www.egov.ufsc.br/portal/conteudo/import%C3%A2ncia-dos-recursos-tecnol%C3%B3gicos-na-sociedade-contempor%C3%A2nea>>. Acesso em: 10 fev. 2020.

Forma Certa. Você sabe o que é flat design?. Forma Certa, 2018. Disponível em: <<http://www.formacerta.com.br/blog/voce-sabe-o-que-e-flat-design/>>.

Virgulino, C. S. O ensino da matemática na educação infantil. Web Artigos, 2014. Disponível em: <<https://www.webartigos.com/artigos/o-ensino-da-matematica-na-educacao-infantil/119953/>>. Acesso em: 24 fev. 2020.

WAZLAWICK, Raul Sidnei. Engenharia de Software: Conceitos e Práticas. Rio de Janeiro: Elsevier, 2013.