

**UNIVERSIDADE ESTADUAL DO MARANHÃO – UEMA
CENTRO DE CIÊNCIAS TECNOLÓGICAS
CURSO DE ENGENHARIA DE COMPUTAÇÃO**

VINICIUS GUILHERME RIBAS DE OLIVEIRA

**TABUADA NO ESPAÇO: DESENVOLVIMENTO DE UM JOGO DIGITAL EDUCATIVO
PARA DISPOSITIVOS MÓVEIS**

**São Luís, MA
2017**

VINICIUS GUILHERME RIBAS DE OLIVEIRA

**TABUADA NO ESPAÇO: DESENVOLVIMENTO DE UM JOGO DIGITAL EDUCATIVO
PARA DISPOSITIVOS MÓVEIS**

Trabalho de Graduação apresentado ao Curso de Engenharia de Computação da Universidade Estadual do Maranhão (UEMA), como requisito para obtenção do grau de Bacharel em Engenharia de Computação.

Orientador: Dr. REINALDO DE JESUS DA SILVA

São Luís, MA

2017

Oliveira, Vinicius Guilherme Ribas de.

Tabuada no espaço: desenvolvimento de um jogo digital educativo para dispositivos móveis / Vinicius Guilherme Ribas de Oliveira. – São Luís, 2017.

62 f.

Monografia (Graduação) – Curso de Engenharia de Computação, Universidade Estadual do Maranhão, 2017.

Orientador: Prof. Dr. Reinaldo de Jesus da Silva

1. Jogos. 2. Educação. 3. Matemática. 4. Aplicativo. I. Título.

CDU:519.837:37.018.43

VINICIUS GUILHERME RIBAS DE OLIVEIRA

**TABUADA NO ESPAÇO: DESENVOLVIMENTO DE UM JOGO DIGITAL EDUCATIVO
PARA DISPOSITIVOS MÓVEIS**

Trabalho de Graduação apresentado ao Curso de Engenharia de Computação da Universidade Estadual do Maranhão (UEMA), como requisito para obtenção do grau de Bacharel em Engenharia de Computação.

Trabalho aprovado. São Luís, MA, 25 de janeiro de 2017:

Dr. REINALDO DE JESUS DA SILVA
Orientador

Dr. LUÍS CARLOS COSTA FONSECA
Primeiro membro

Esp. MARCOS DOS SANTOS SOUZA
Segundo membro

Dedicatória

Dedico este trabalho primeiramente a Deus, sem ele não seria possível concluir essa etapa, aos meus pais, familiares e amigos por todo o apoio que me deram em toda essa jornada.

Agradecimentos

Aos meus pais, por me apoiarem todos esses anos e me fornecerem condições para ter chegado até aqui.

Ao meu orientador, que dedicou parte de seu tempo para acompanhar esta pesquisa e por acrescentar bastante valor a mesma.

Aos meus irmãos, por todos os debates sobre a vida, jogos e filmes.

A Paula Sousa, por ter feito esse magnífico trabalho de designer no jogo que o tornou tão impressionante.

A Ana Paula, por toda ajuda e empenho como amiga e representando os alunos à frente do DA.

A minha noiva, Vanessa Memoria, que me apoiou, me ouviu e me confortou nos momentos mais difíceis.

*"As pessoas não sabem o que querem,
até mostrarmos a elas."
(Steve Jobs)*

Resumo

Os métodos de ensino estão em constante evolução, pesquisadores dessa área buscam sempre novas metodologias e conceitos para melhorar e auxiliar o processo de aprendizagem. Uma dessas novas ferramentas é a utilização de jogos digitais educativos vinculados ao processo de ensino-aprendizagem para tornar o ensino mais divertido, entretanto, todo o processo de concepção precisa estruturar o aprendizado a partir de marcações intermediárias embasadas em cada ciência ligada às disciplinas, pedagogia e psicologia, desenvolvendo o cognitivo a partir de cada etapa vivida e compartilhada no jogo.

Neste trabalho foram utilizados conceitos pedagógicos, aliado com técnicas de modelagem para definir a arquitetura do sistema e desenvolver um protótipo do jogo “Tabuada no Espaço” em celulares com o sistema operacional Android. Esse jogo tem o objetivo de auxiliar no aprendizado de operações aritméticas dos tipos: adição, subtração, multiplicação e divisão, voltado para crianças nos primeiros anos do ensino fundamental. Para validar o protótipo desenvolvido, foram realizados testes com professores, possibilitando verificar que o aplicativo tem bons resultados nos quesitos de adaptação, diversão e engajamento.

Palavras-chaves: Aplicativo, Ensino Fundamental, Aprendizagem, Jogos.

Abstract

The teaching methods are constantly evolving, researchers in this area are always looking for new methodologies and concepts to improve and assist the learning process. One of these new tools is the use of digital educational games linked to the teaching-learning process to make teaching more fun. However, the entire design process needs to structure learning based on intermediate markings based on each discipline-related science, pedagogy and psychology, developing the cognitive from each step lived and shared in the game.

In this work we used pedagogical concepts, allied with modeling techniques to define the system architecture and to develop a prototype of the game "Tabuada no Espaço" on mobile phones with the Android operating system. This game aims to aid in the learning of arithmetic operations of the types: addition, subtraction, multiplication and division, aimed at children in the first years of elementary school. To validate the developed prototype, tests were carried out with teachers and children, making it possible to verify that the application has good results in terms of adaptation, fun and engagement.

Key-words: Application, Educational, Learning, Games.

Sumário

1	INTRODUÇÃO	11
1.1	Metodologia	12
1.2	Objetivo Geral	12
1.2.1	Objetivos Específicos	12
1.3	Apresentação do trabalho	12
2	FUNDAMENTAÇÃO TEÓRICA	14
2.1	A matemática e o lúdico	14
2.2	Jogos digitais	15
2.3	Importância dos jogos digitais na educação	16
2.4	Engines	17
2.4.1	Corona SDK	18
2.5	Lua	19
2.6	Android	20
3	TRABALHOS RELACIONADOS	22
3.1	O homem que calculava	22
3.2	Labirinto do Rato	23
3.3	Number Count	24
3.4	Comparação das ferramentas	25
4	ARQUITETURA DO SISTEMA	27
4.1	Tabuada no Espaço	27
4.2	Definição do sistema	29
4.2.1	Plataforma	29
4.2.2	Engine e Linguagem de programação	29
4.3	Modelagem do Sistema	29
4.3.1	Casos de uso	29
4.3.2	Diagrama de classes	30
4.3.3	Fluxogramas	31
4.3.3.1	Tratamento de colisão	31
4.3.3.2	Fluxo da fase	32
5	PROTOTIPAGEM DO APLICATIVO TABUADA NO ESPAÇO	34
5.1	História do jogo	34
5.2	Elementos da tela de jogo	35
5.2.1	Tela de escolha de níveis	35
5.2.2	Tela de jogo	36

5.3	Teste	38
6	CONCLUSÃO	39
6.1	Trabalhos Futuros	40
	Referências	41
	Apêndices	43
	APÊNDICE A – CÓDIGO DO MAIN.LUA	44
	APÊNDICE B – CÓDIGO DO MENU.LUA	45
	APÊNDICE C – CÓDIGO DO LEVEL.LUA	46
	Anexos	60
	ANEXO A – QUESTIONÁRIO APLICADO AOS PROFESSORES . .	61

1 INTRODUÇÃO

Com a popularização dos celulares e computadores e a facilidade de acesso à informação por uma grande parte da população, sendo que esse acesso se dá cada vez mais cedo, acabou surgindo um novo tipo de usuário, as crianças, que não estão mais limitadas ao uso do desktop em casa ou na escola, tendo seus próprios aparelhos celulares. Nessa realidade, em que crianças têm tecnologias em suas mãos, capazes de diverti-las em qualquer lugar que estejam, surgiu a ideia de unir a diversão ao aprendizado, os jogos digitais educacionais nasceram dessa busca por novas ferramentas e métodos para estabelecer conexão com os planejamentos educacionais.

Alexandre e Sabbatini (2013) afirmam que nunca se questionou tantos os ganhos em processos de aprendizagem quanto os que usam jogos digitais educacionais. Um dos motivos para o receio da aceitação desse modelo é a cultura de que o jogo digital está ligado apenas à diversão, lazer e entretenimento. Apesar disso, de acordo com Moran (2015) o uso de tecnologias digitais como celulares, tablets e computadores trazem novas possibilidades e grandes desafios, pois o aprendizado pode ocorrer também em locais diferentes do ambiente escolar.

De acordo com Schneider (2007) ao iniciar sua vida escolar, a criança não aprende somente a ler e escrever, mas também a conhecer números e fazer contas. Essa necessidade de ensinar Matemática é defendida com argumentos de que desenvolve o lado cognitivo da criança. Pais (2013) ressalta que apesar dessa alegação, isso não garante que os objetivos propostos sejam alcançados, pois o ensino da Matemática ao longo dos últimos anos está embasado numa prática de repetição e memorização.

Na busca de novas metodologias no ensino da matemática nos primeiros anos do ensino Fundamental, encontrou-se nos jogos, benefícios que pudessem estimular, desenvolver e provocar o raciocínio de uma criança por meio da interatividade aliada aos recursos tecnológicos, favorecendo o processo de ensino e aprendizagem. Machado (2012) defende que o ato de jogar pode explorar diferentes situações para inúmeros propósitos, a ação por meio do jogo pode ser utilizada na aprendizagem de conteúdos matemáticos de uma maneira mais lúdica.

Mattar (2010) comenta que se faz necessário entender como a sistemática de jogos pode auxiliar professores no ensino da matemática. O aprendizado por meio dos jogos digitais ocorre de maneira dinâmica, observa-se que os alunos assumem uma postura de interação no processo de aprendizagem, agindo de forma proativa, tendo a liberdade de escolher como, quando e por que aprendem.

Sobre os conceitos levantados se faz o seguinte questionamento: como auxiliar no processo de aprendizagem da matemática utilizando um jogo digital que alia elementos de uma mecânica divertida com atividades que exploram os conceitos de premiação, promoção e bonificação?

Este trabalho busca produzir um jogo para auxiliar o aprendizado de operações matemáticas do tipo adição, subtração, multiplicação e divisão, proporcionando um ambiente interativo

e divertido para ajudar no ensino durante os primeiros anos da escola, pretendo assim produzir conhecimento para área de jogos educacionais. Podendo se tornar um suporte para professores começarem a explorar com seus alunos as primeiras operações contidas na tabuada.

Um protótipo é utilizado para validar e testar a modelagem de um software, feita durante a fase de planejamento e conceituação. Dessa forma, para complementar e verificar se esse trabalho pode alcançar o resultado esperado, será desenvolvido uma prévia do aplicativo utilizando técnicas de prototipagem e ferramentas de criação de jogos, como a engine, conhecida como motor de jogo. Esse protótipo será testado em escolas com a supervisão de professores, aplicando questionários com os docentes para analisar suas impressões e opiniões sobre a ferramenta.

1.1 Metodologia

Neste trabalho será utilizado em um primeiro momento uma abordagem comparativa, sendo levantados pesquisas na área de jogos digitais educativos, com intuito de analisar a abordagem e resultados para comparar com o objetivo proposto deste trabalho, essas comparações serão feitas seguindo critérios pré-estabelecidos de uma forma qualitativa.

Posterior para testar a modelagem do aplicativo e verificar se o resultado do trabalho pode ser alcançado, será desenvolvido um protótipo para ser aplicados em escolas. Nessas escolas serão aplicados questionários com 3 professores de 3 turmas diferentes com perguntas de múltiplas escolhas, para que o resultado da pesquisa possa ser mensurado.

1.2 Objetivo Geral

Esta pesquisa tem como objetivo desenvolver um aplicativo móvel para o sistema operacional Android com o intuito de auxiliar na aprendizagem de matemática para crianças nos primeiros anos do ensino fundamental.

1.2.1 Objetivos Específicos

- Estudar os mecanismos de desenvolvimento de aplicativos para o sistema operacional Android;
- Definir conceitos do jogo;
- Modelar aplicativo utilizando linguagem UML;
- Desenvolver prototipagem do aplicativo.

1.3 Apresentação do trabalho

Esta pesquisa está dividida em 5 capítulos, dispostos da seguinte maneira a saber:

- O capítulo 1 aborda a introdução da pesquisa. Ela aponta os objetivos gerais, específicos e a motivação que impulsionou o desenvolvimento do projeto;
- O capítulo 2 trata da fundamentação teórica que serve de base de conhecimento para o desenvolvimento do projeto. Durante esse capítulo os conceitos de engine, jogos e metodologias educacionais serão abordados;
- O capítulo 3 relaciona trabalhos na mesma área de pesquisa e com objetivos semelhantes para fazer uma comparação com a proposta desse trabalho;
- O capítulo 4 trata da arquitetura do sistema, nesse capítulo é definido os conceitos do jogo, o sistema operacional em que será usado, a engine selecionada e sua modelagem com linguagem UML;
- O capítulo 5 trata da prototipagem do jogo. Durante esse capítulo será apresentado a história do jogo, explicações de sua interface, como serão divididas as fases e os resultados dos testes com o protótipo;
- O capítulo 6 trará as conclusões e resultados da prototipagem, com também propostas de trabalhos futuros relacionados ao projeto.

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo serão abordados os conteúdos que giram em torno do projeto desta monografia. Eles proporcionam uma base sólida e um bom alicerce para sustentar a ideia que o projeto apresenta. Serão abordados alguns conceitos pedagógicos como o lúdico, seguindo pelo levantamento do histórico dos jogos educacionais, avançando posteriormente para conceitos sobre as tecnologias utilizadas como engines, a linguagem de programação Lua e o Android.

2.1 A matemática e o lúdico

Machado (2011) afirma que jogar faz parte do crescimento de uma criança, essa experiência possibilita a ela vivenciar situações e desempenhar papéis diferentes, toda criança experimenta esse tipo vivência, não importa a cultura. Lorenzato (2011) e Pais (2013) ressaltam que antes de irem para escola, as crianças convivem com situações matemáticas de contar, comparar, juntar, retirar, medir, distribuir e medir. A brincadeira, de forma especial o jogo, possibilita às crianças experimentarem situações de convívio com numerais, contagem e operações aritméticas, de maneira verbal ou escrita.

Na educação matemática, para tentar mudar a imagem de que é complicado ensinar e aprender matemática, o lúdico tem sido objeto de estudo por grupos de pesquisadores, na figura 1 mostra um exemplo de um atividade lúdica que aborda conceitos da matemática. Esses estudos buscam pela consolidação de novas estratégias para auxiliar no processo de ensino e aprendizagem.

Figura 1 – Atividade lúdica com conceitos matemáticos.



Fonte: www.trabalhosdaprofivani.blogspot.com.br, 2012

Smole, Diniz e Candido (2007) afirmam que o ato de jogar, unido com a mediação do docente pode propiciar situações de aprendizagem, pois a experiência em novas partidas pela criança, resulta em ações como a de resolução de problemas, investigação da resposta e descobrimento da melhor alternativa. O processo de aprendizagem da matemática acontece

de maneira significativa para o aluno, quando as situações provocam raciocínio, insistência e experimentação, de maneira prazerosa, a ação de jogar permite explorar esse cenário.

2.2 Jogos digitais

A história dos jogos digitais está profundamente ligada ao desenvolvimento do computador, dessa forma é preciso relacionar essa evolução aos principais projetos e experimentos computacionais. Aranha (2004) afirma que é possível apontar como marco inicial dessa história, o desenvolvimento pelos nazistas de uma máquina que criptografava mensagens durante a Segunda Guerra Mundial e o trabalho de Alan Turing que culminou na construção de um dispositivo que podia “imitar” o cérebro humano quanto ao processamento de milhares de alternativas possíveis a partir de uma única variável.

Kent (2001) afirma que as tecnologias militares estão diretamente ligadas na criação dos jogos eletrônicos, tanto que o primeiro jogo digital que se tem notícia foi desenvolvido pelo William Higinbotham, um físico americano em 1958 para mostrar o desenvolvimento e avanços tecnológicos para a população dos Estados Unidos. Para demonstração foi criado o "Tennis Programming", onde em um osciloscópio, conforme mostra a figura 2, se jogava uma partida de tênis interativa que ficou conhecida como “Tennis for Two”.

Figura 2 – Jogo "Tennis for Two" que rodava em um osciloscópio.



Fonte: www.m-e-g-a.org/research-education/research/t42-tennis-for-two , 2011

Aranha (2004) ressalta que outro marco importante foi o projeto que se tornou o primeiro console, com o nome de Brown Box, em 1971 a Magnabox desenvolveu o Odyssey 100, que se tornou o primeiro console de jogos eletrônicos fabricado, esse dispositivo usava placas de circuito impresso externas que controlavam a movimentação das barras retangulares na tela.

A tecnologia na área dos jogos evoluiu muito nos últimos 50 anos, entretanto não foi apenas o aspecto tecnológico que evoluiu, com o passar dos anos conceitos de como um jogo digital deve ser, foram se refinando. Lucchese e Ribeiro (2012) revelam que um dos principais conceitos de um jogo digital é a determinação que dentro de um jogo ações e decisões são limitadas por um sistema baseado em regras e por um universo, que é regido de acordo com sua

programação. O universo contextualiza e reflete as ações e decisões do jogador, fornecendo um ambiente adequado para desenvolvimento da narrativa do jogo. As regras fornecem desafios a fim de limitar, dificultar ou impedir que o jogador de alcançar os objetivos estabelecidos.

2.3 Importância dos jogos digitais na educação

O desenvolvimento de jogos educacionais não é uma área nova, esse tipo de ferramenta de aprendizagem está nas salas de aulas faz tempo, jogos físicos que estimulam as crianças a pensarem e interagirem, exemplo disso são os jogos da memória, de tabuleiro e caça palavras. Entretanto existe uma grande resistência quando se trata de jogos digitais, conforme citado anteriormente, o questionamento é muito grande quando se tenta mensurar os ganhos na aprendizagem.

Alexandre e Sabbatini (2013) argumentam que muito já foi feito na relação entre jogos e a escola, exemplos disso são a infinidade de jogos disponíveis podendo ser até gratuitos e que tem como o objetivo ensinar alguma coisa, porém muito desses jogos continuam não sendo o suficiente para motivar ou engajar os alunos, um dos motivos dessa falta de atração de jogos educacionais é o fato de terem que cumprir o papel de ensinar algum conteúdo específico ao jogador, podendo ser comparado a uma sala de aula tradicional, mas através do computador.

Dessa forma Falkembach (2007) afirma que para produzir um jogo educacional bem estruturado é preciso que ele tenha as seguintes características: ser atrativo, agradável e fácil de usar. O aluno precisa entender de forma fácil os comandos básicos e as opções para navegar dentro da aplicação e jogar sem dificuldades.

Huizinga (2000) afirma que o jogo tem que ser uma atividade voluntária realizada dentro de um tempo e espaço limitado, através de regras que delimitam ações e o seu fim, onde estão presentes sentimentos de tensão e alegria, entretanto sem causar muita punição ao jogador que fracassa, para não diminuir sua motivação.

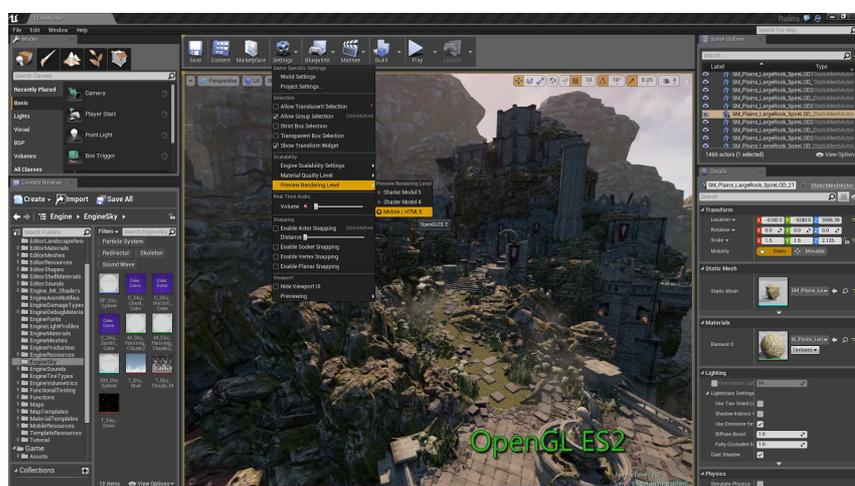
Constantemente na escola o erro está ligado diretamente ao fracasso, para Fiorentini e Lorenzato (2006), o erro é resultado da tentativa do aluno em envolver-se no processo de aprendizagem, utilizando dos seus conhecimentos prévios e significados sobre o que é ensinado e aprendido na escola. Na dimensão lúdica, o erro recebe uma conotação natural, Mattar (2010) lembra que quando uma criança experimenta um jogo digital, o custo do fracasso é minimizado, quando um jogador perde, pode recomeçar a partir da fase em que se encontrava antes da derrota, de forma rápida. Desta forma o fracasso é encarado como uma maneira de aprender em função dos erros cometidos e, numa oportunidade futura, conseguir vencer.

Smole, Diniz e Candido (2007) enaltecem que com experiência das partidas desenvolvem-se antecipações e previsões nas atitudes a serem tomadas durante o acontecimento das disputas, finalizando com a realização de melhores jogadas com os conhecimentos adquiridos previamente, possibilitam a aquisição de novas ideias e conhecimentos.

2.4 Engines

Engine, também conhecido como Motor de jogo é uma ferramenta de desenvolvimento que serve para abstrair parte do processo de construção de um jogo. Clua e Bittencourt (2005) relacionam uma engine com um motor de carro, onde ao ligar o veículo, o motorista coloca o motor em funcionamento e se desloca com ele, sem precisar saber do funcionamento do processo mecânico. O princípio de um motor de jogo é o mesmo, durante o desenvolvimento de um jogo parte do processo é simplificado com a ajuda dessa ferramenta. A figura 3 mostra o ambiente de desenvolvimento da engine Unreal, nesse ambiente é possível controlar as texturas, objetos e ambiente.

Figura 3 – Tela de desenvolvimento da engine Unreal.



Fonte: www.forums.unrealengine.com, 2015

A engine em tempos de execução fica responsável pela compatibilidade com o hardware gráfico, controla os modelos que serão renderizados, gerencia as entradas do jogador e fica responsável pelo processamento de baixo nível. Existem diversas engines no mercado de desenvolvimento de jogos, entretanto é comum encontrar características parecidas entre elas, como por exemplo, suas ferramentas e componentes que são bem parecidas na finalidade. Clua e Bittencourt (2005) citam 4 componentes muito comuns entre as engines:

- **Engine Core:** Considerado como o coração da engine, esse componente é responsável por executar a manipulação das fases e os objetos, renderização das cenas, entre outras coisas;
- **Engine SDK:** É o kit de desenvolvimento da engine, utilizada para personalizar o seu funcionamento de acordo com a necessidade;
- **Level Editors:** Tem as funções de importar modelagens de outras ferramentas e atribuir uma programação;

- **Linguagens Script:** A maioria das engines utilizam linguagens de programação script como Python e Lua para o desenvolvimento da lógica do jogo.

2.4.1 Corona SDK

A engine Corona SDK foi desenvolvida pela empresa Corona Labs (figura 4) e foi lançada em 2010. Atualmente o Corona está na sua versão 2016.2987 e gera aplicações para iOS, Android, Kindle, Windows Phone 8, Apple TV, Android TV, OS X e Windows Desktops (CORONA, 2016).

Figura 4 – Logo da empresa Corona Labs.



Fonte: www.coronalabs.com, 2017

O motor de jogo escolhido reduz significativamente o tempo de programação, pois utiliza linguagens de script para o desenvolvimento, aliado com suas bibliotecas completas e de fácil utilização, a principal linguagem usada no Corona é o Lua, que se tornou muito popular entre os desenvolvedores de jogos pela facilidade na construção do código, conforme mostra o exemplo da figura 5.

Figura 5 – Exemplo da facilidade de importação e utilização das bibliotecas do Corona SDK.

```
local physics = require("physics")
local bola = display.newCircle( 15, 15, 15 )
physics.addBody(bola)
```

Fonte: do autor, 2017

O Corona SDK se destaca também na facilidade de gerar uma aplicação para vários sistemas operacionais, isso é, utilizando um mesmo código, pode se criar, por exemplo, um aplicativo para iOS e Android. Davila (2012) alerta que ao criar jogos para iOS é preciso utilizar um dispositivo Macintosh, pois mesmo o Corona possuindo uma versão para o Windows, não existe o SDK do iOS para esse sistema operacional.

A licença do Corona SDK para pequenos desenvolvedores é gratuita, apenas para empresas um valor mensal é cobrado, o preço varia da licença para uma pequena empresa e de uma grande. A diferença entre a versão gratuita e a de empresa é a restrição de algumas bibliotecas mais avançadas.

As bibliotecas do Corona são acessadas através de uma importação via código, conforme mostra a figura 5. As bibliotecas mais utilizadas durante o desenvolvimento de um jogo com essa engine:

- **Composer:** Trata da composição da tela do jogo, sendo responsável por importar os objetos necessários, mudar os elementos da tela e apagar objetos de uma outra tela já utilizada;
- **Object:** Utilizada para instanciar um objeto e adicionar suas características como tamanho, imagem e etc;
- **Physics:** Responsável por adicionar e tratar a física do jogo, através dessa biblioteca serão adicionados elementos como a gravidade, tratamento de colisão e características do corpo (peso, escala e velocidade de interação);
- **Event:** Através dessa biblioteca pode se definir como será a resposta de um objeto ou do sistema de acordo com um evento que ocorre, como por exemplo um conflito de física, um toque ou até mesmo no início ou final do jogo.

2.5 Lua

Lua (figura 6) é uma linguagem de programação que nasceu com o objetivo de estender aplicações, tem uma sintaxe simplificada para programação procedural e semântica extensível, sendo gratuita sob a licença MIT e tem seu código aberto. Essa linguagem é tipada dinamicamente, sendo interpretada a partir de bytecodes para uma máquina virtual com gerenciamento automático de memória com coleta de lixo incremental (LUA, 2017).

Figura 6 – Símbolo da linguagem Lua.



Fonte: www.lua.org , 2017

A linguagem foi criada no Grupo de Tecnologia em Computação Gráfica da PUC-RIO (Tecgraf) em 1993 (LUA, 2017). Segundo DeLoura (2001) o Lua é amplamente utilizado no desenvolvimento de jogos por seu ótimo desempenho em comparação a outras linguagens.

Davila (2012) explica que a linguagem Lua possui três tipos de variáveis: locais, globais e campos de tabelas. O que diferencia uma variável global de uma local é o nível de acesso que os elementos tem sobre aquela variável, é possível adicionar um bloco de código e manter a validade de uma variável local somente dentro desse bloco. Variáveis locais são iniciadas com o comando local, a ausência dessa instrução, torna a variável global. Os campos de tabela são os elementos das tabelas, os índices podem ser números inteiros ou strings, quando utilizado com números inteiros, a tabela é tratada como uma matriz, caso seja utilizado strings ela passa ser tratada como uma propriedade.

2.6 Android

O Android (figura 7) é um sistema operacional para dispositivos móveis desenvolvido pela Open Handset Alliance, liderada pelo Google. Segundo Pereira e Silva (2009) esse sistema operacional foi desenvolvido para possibilitar aos desenvolvedores criar aplicações para dispositivos móveis que podem utilizar a maioria dos recursos que um smartphone pode oferecer. Através de um aplicativo no Android é possível efetuar chamadas, utilizar a câmera e enviar mensagens de texto. Com essa liberdade, o Android permite que os novos aplicativos adaptem e evoluam funcionalidades nativas.

Figura 7 – Símbolo do Android.



Fonte: www.android.com, 2016

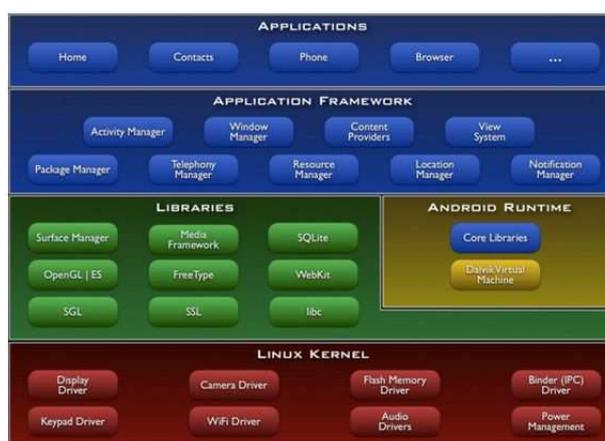
O crescimento da popularidade entre os usuários de smartphone foi o que incentivou nos últimos anos o crescimento da comunidade de desenvolvimento, em 2013 um estudo levantado pela Analytics (2014) afirma que o Android assumiu quase 79% da fatia global de unidades enviadas para comercialização. Crescimento que aumentou durante os anos, sendo que

segundo a consultoria Gartner (2016), os aparelhos com Android no segundo trimestre de 2016 representaram mais de 86,2% dos celulares vendidos.

Esse sistema operacional é baseado no Linux e é composto por um conjunto de ferramentas que atuam em todas as fases de desenvolvimento do projeto. Pereira e Silva (2009) afirmam que apesar de ser baseado no Linux, o Android não pode ser considerado como um, pois não possui o windowing system nativo (componentes de GUI) e não possui alguns dos conjuntos padrões como nas distribuições Linux.

A arquitetura do Android é dividida em 4 partes, como mostrado na figura 8. O núcleo linux ou conhecido também por kernel linux, é a base do sistema, sendo responsável pelos drives de hardware e redes (Bluetooth, USB, GPS e etc), também gerencia o sistema de arquivos e processamento. Já as bibliotecas nativas (Libraries) são recursos que podem ser utilizados no desenvolvimento de aplicações. A parte de Frameworks são serviços e bibliotecas que tem como objetivo facilitar no desenvolvimento de aplicativos e jogos. A última parte é a camada de aplicação, local onde ficam localizados os aplicativos e jogos desenvolvidos em Java.

Figura 8 – Diagrama que representa a arquitetura do Android.



Fonte: www.developer.android.com, 2016

O kit de desenvolvimento que disponibiliza as ferramentas e APIs necessárias para desenvolver aplicações no Android utilizando linguagem Java é o Android SDK, toda aplicação precisa ser assinada com um certificado cuja chave privada é mantida pelo seu desenvolvedor. Este certificado é utilizado somente para estabelecer relações de confiança entre as aplicações (PEREIRA; SILVA, 2009).

Existem outras formas de desenvolver para Android sem contato direto com o SDK, as engines são exemplos disso. Utilizando essas ferramentas pode-se abstrair o contato com as bibliotecas do Android e desenvolver um aplicativo utilizando outras linguagens, além do Java. Esses programas utilizam o SDK como base para que os desenvolvedores não se preocupem com isso em tempo de desenvolvimento da aplicação.

3 TRABALHOS RELACIONADOS

Neste capítulo serão abordados alguns trabalhos encontrados na literatura que buscaram desenvolver um jogo educacional. O objetivo é entender e avaliar abordagens diferentes na área de desenvolvimento de ferramentas de auxílio de aprendizagem. Os critérios para escolha dos trabalhos foram:

- O jogo como produto final do trabalho;
- Para sistemas operacionais Android;
- Ensino de lógica ou matemática;
- Para alunos do ensino fundamental.

3.1 O homem que calculava

O Jogo "O homem que calculava" desenvolvido para celulares com o sistema operacional Android tem como o objetivo apresentar de forma diferente alguns conceitos matemáticos vistos em sala de aula. Neto (2013) conta que o enredo do seu jogo é baseado na obra de Malba Tahan, pseudônimo do professor de matemática Julio Cesar de Mello e Souza, e foi escolhido por ser uma história que aborda problemas matemáticos. A figura 9 mostra o menu inicial do jogo.

Figura 9 – Tela de entrada do jogo "O homem que calculava".



Fonte: Neto (2013)

O jogo foi dividido em seis estágios e tem nível de dificuldade crescente, cada estágio contém um desafio e uma pequena história que envolvem conceitos matemáticos. No primeiro nível, por exemplo, conta a história de Beremiz que está viajando com um amigo pelo deserto, durante essa viagem eles encontram três irmãos que precisam dividir uma certa quantidade de camelos que receberão de herança, porém existe uma regra para essa divisão, o primeiro filho

receberá metade da herança, o segundo filho receberá um terço e o terceiro filho um nono da herança. A figura 10 mostra os elementos da primeira fase.

Figura 10 – Tela da primeira fase do jogo "O homem que calculava".



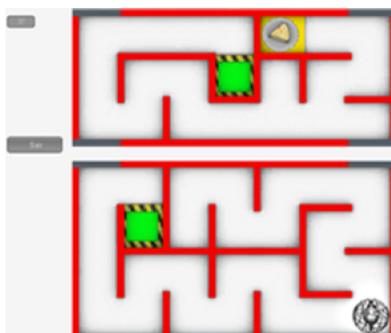
Fonte: Neto (2013)

O aluno terá que resolver essa divisão de forma correta, caso erre o sistema exibirá uma mensagem que a resposta está errada e o estágio inicia novamente, caso tenha acertado, o sistema exibirá uma mensagem e encaminha o jogador para tela de conceito para responder qual o conteúdo que foi abordado durante a fase.

3.2 Labirinto do Rato

Desenvolvido para celulares com o sistema operacional Android, o jogo "Labirinto do Rato" tem o objetivo de estimular a criatividade, concentração e estratégias de raciocínio, uma vez que o jogador deve memorizar os caminhos já percorridos no labirinto e ter uma localização baseada na estrutura do mesmo. A figura 11 mostra uma das fases do jogo.

Figura 11 – Tela da fase do jogo "Labirinto do Rato".



Fonte: Silva, Nobrega e Jacob (2011)

O projeto foi desenvolvido para utilizar o acelerômetro do dispositivo móvel para o controle dos movimentos, em outras palavras, para movimentar o personagem dentro do jogo

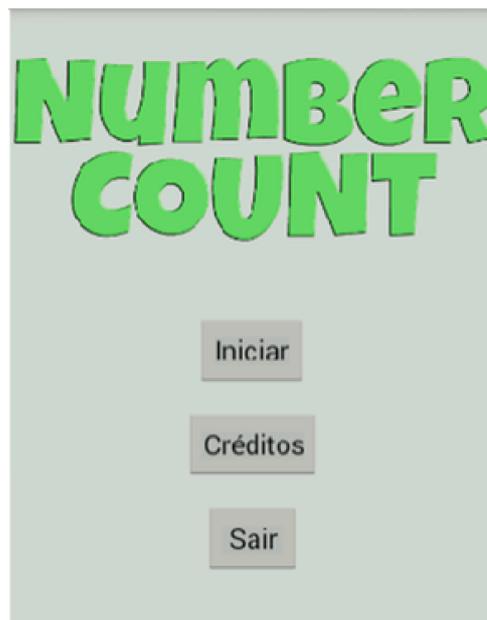
é preciso movimentar o dispositivo fisicamente. O personagem é um rato em uma esfera que precisa alcançar o final sem tocar nos obstáculos.

Silva, Nobrega e Jacob (2011) descrevem que o jogo possui 4 níveis, cada estágio possui um caminho diferente e plataformas coloridas que transportam o jogador para outro ponto mais próximo da saída, sendo que quanto mais rápido o usuário chegar no final mais pontos ganha, essa técnica é utilizada para desafiar o jogador e aumentar o número de repetições e interações com o jogo.

3.3 Number Count

O "Number Count" foi desenvolvido para o auxílio na aprendizagem da matemática básica, esse jogo possibilita o aluno a realização de operações matemáticas de soma e subtração, sem exigir inicialmente o conhecimento da representação gráfica dos numerais. Almeida et al. (2014) explicam que tem como expectativa que com o progresso das fases os alunos construam conhecimento para associar a quantidade apresentada ao numeral, para poderem realizar as operações que o jogo dispõe. A figura 12 mostra o menu inicial do jogo.

Figura 12 – Tela de entrada do jogo "Number Count".



Fonte: Almeida et al. (2014)

O jogo é dividido em três fases que ambientam um supermercado, o local foi escolhido para construir um ambiente lúdico e comum na vida da criança, estimulando e aumentando seu interesse pelo jogo. Na primeira fase do jogo deverá ser realizadas operações com objetos representados por maçãs, a figura 13 mostra que essa operação pode ser feita através da contagem.

Figura 13 – Tela da primeira fase do "Number Count".



Fonte: Almeida et al. (2014)

Para a criança é mostrado dois conjuntos com uma certa quantidade de maçãs, o jogador terá que fazer a conta e arrastar a quantidade certa para um elemento agrupador. Durante esse nível o jogo propõe a capacidade do jogador de entender primeiro a quantidade para depois precisar relacionar com os números. Conseguindo absorver conceitos de como fazer uma operação aritmética por partes.

3.4 Comparação das ferramentas

A escolha dos trabalhos relacionados levou em consideração as necessidades levantadas por esta pesquisa. A tabela 1 apresenta um comparativo entre as características dos trabalhos escolhidos e o "Tabuada no Espaço" que será apresentado no próximo capítulo.

O "Labirinto do Rato" é o que tem a proposta de jogo diferente dos outros, buscando estimular a lógica e concentração do jogador ao tentar escapar do labirinto, também possui outro ponto positivo por estimular a coordenação motora do jogador, pois utiliza o acelerômetro para mover o personagem.

Pode-se observar também que a proposta inicial é criar um aplicativo para o Android, entretanto no desenvolvimento do jogo será utilizado técnicas que tornam possível que o mesmo código do jogo feito para Android seja compatível com outras plataformas, como por exemplo, o Windows Phone e iOS.

"O homem que calculava" apesar de ter uma proposta bem interessante ensinando conceitos matemáticos de uma forma imersiva, não se encaixa na proposta quando o objetivo

Tabela 1: Tabela de comparação entre os aplicativos.

PROJETO	Celulares com sistema operacional Android	Apresenta conceitos matemáticos	Auxiliar no ensino de operações aritméticas (soma, subtração, multiplicação e divisão)	Voltado para o público Infantil (primeiros anos do ensino fundamental)
O homem que calculava	x	x	x	-
Labirinto do Rato	x	-	-	x
Number count	x	x	-	x
Tabuada no Espaço	x	x	x	x

Fonte: do autor, 2017

é voltado para um público mais infantil. Nesse ponto o "Number Count" se encaixa muito bem, pois ensina de uma forma que abstrai o uso dos operadores, entretanto como a proposta é o ensino das operações que se iniciam na soma e progridem para subtração, multiplicação e divisão, acaba não contemplando todo o cenário que se busca do jogo proposto. Depois da análise das ferramentas relacionadas, foi possível constatar que o desenvolvimento do "Tabuada Espacial" se tornou uma proposta válida.

4 ARQUITETURA DO SISTEMA

Definir a arquitetura de um sistema se tornou uma etapa essencial no desenvolvimento de um software, aplicação ou jogo. Sistemas sem arquitetura ou planejamento demoram mais tempo para ficarem prontos, ou custam muito mais do que previsto, existem casos em que o desenvolvimento é mais rápido, mas o custo posterior com manutenção e resolução de problemas acaba sendo maior. Esse novo cenário é a tendência de todos os desenvolvedores que buscam o sucesso a longo prazo.

Neste capítulo serão utilizadas técnicas de modelagem para definir a arquitetura do projeto "Tabuada no Espaço", os diagramas seguirão técnicas baseadas a UML (Unified Modeling Language).

4.1 Tabuada no Espaço

"Tabuada no Espaço" é um jogo com o objetivo de auxiliar no aprendizado de matemática de uma forma lúdica voltada para crianças nos primeiros anos do ensino fundamental. O jogo tem a proposta de ensinar operações aritméticas de uma forma divertida. Para responder as perguntas feitas, o jogador deve movimentar o personagem na vertical para desviar dos inimigos e pegar a resposta adequada. Em cada nível o jogador recebe 5 chances para errar, sendo que os erros são representados por corações, como mostra a figura 14. Cada fase possui 10 operações aritméticas para serem respondidas, para concluir o nível o jogador deve acertar a maior parte das questões.

Figura 14 – Simulação do jogo sendo executado no Corona SDK.

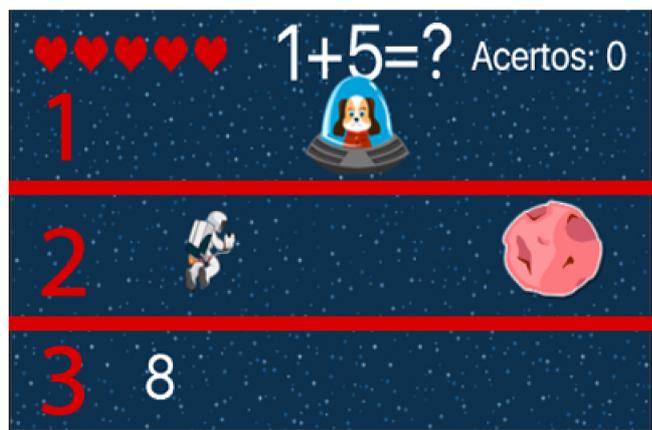


Fonte: do autor, 2017

As perguntas são mostradas na parte superior da tela no modelo "4+5=?". Para responder, o jogador deve "passar" por cima da resposta adequada. Ao responder de forma correta o jogador

recebe uma resposta positiva e é contabilizado 1 ponto, ao errar o jogador é alertado do erro e perde um coração. As fases são divididas horizontalmente em três faixas, essas faixas são os caminhos que o jogador pode se deslocar, a faixa 1 representa o mais alto que o jogador pode "subir" e a faixa 3 é o máximo que ele pode descer, conforme mostra a figura 15.

Figura 15 – Faixas do jogo sendo simuladas no Corona SDK.



Fonte: do autor, 2017

O jogo possui 9 fases, sendo que cada fase terá um tipo de operação para ser resolvida, por exemplo, nas primeiras três fases, o jogador terá que responder operações de soma, a figura 16 mostra as tabuadas que são utilizadas nesses níveis.

Figura 16 – Tabuada de 0 a 9 de adição.

Tabuada da Adição				
0 + 0 = 0	1 + 0 = 1	2 + 0 = 2	3 + 0 = 3	4 + 0 = 4
0 + 1 = 1	1 + 1 = 2	2 + 1 = 3	3 + 1 = 4	4 + 1 = 5
0 + 2 = 2	1 + 2 = 3	2 + 2 = 4	3 + 2 = 5	4 + 2 = 6
0 + 3 = 3	1 + 3 = 4	2 + 3 = 5	3 + 3 = 6	4 + 3 = 7
0 + 4 = 4	1 + 4 = 5	2 + 4 = 6	3 + 4 = 7	4 + 4 = 8
0 + 5 = 5	1 + 5 = 6	2 + 5 = 7	3 + 5 = 8	4 + 5 = 9
0 + 6 = 6	1 + 6 = 7	2 + 6 = 8	3 + 6 = 9	4 + 6 = 10
0 + 7 = 7	1 + 7 = 8	2 + 7 = 9	3 + 7 = 10	4 + 7 = 11
0 + 8 = 8	1 + 8 = 9	2 + 8 = 10	3 + 8 = 11	4 + 8 = 12
0 + 9 = 9	1 + 9 = 10	2 + 9 = 11	3 + 9 = 12	4 + 9 = 13
5 + 0 = 5	6 + 0 = 6	7 + 0 = 7	8 + 0 = 8	9 + 0 = 9
5 + 1 = 6	6 + 1 = 7	7 + 1 = 8	8 + 1 = 9	9 + 1 = 10
5 + 2 = 7	6 + 2 = 8	7 + 2 = 9	8 + 2 = 10	9 + 2 = 11
5 + 3 = 8	6 + 3 = 9	7 + 3 = 10	8 + 3 = 11	9 + 3 = 12
5 + 4 = 9	6 + 4 = 10	7 + 4 = 11	8 + 4 = 12	9 + 4 = 13
5 + 5 = 10	6 + 5 = 11	7 + 5 = 12	8 + 5 = 13	9 + 5 = 14
5 + 6 = 11	6 + 6 = 12	7 + 6 = 13	8 + 6 = 14	9 + 6 = 15
5 + 7 = 12	6 + 7 = 13	7 + 7 = 14	8 + 7 = 15	9 + 7 = 16
5 + 8 = 13	6 + 8 = 14	7 + 8 = 15	8 + 8 = 16	9 + 8 = 17
5 + 9 = 14	6 + 9 = 15	7 + 9 = 16	8 + 9 = 17	9 + 9 = 18

Fonte: <http://www.ativeatabuada.com.br/>, 2016

Cada fase possui dez perguntas, os questionamentos são gerados de forma aleatória de acordo com a tabuada correspondente, por exemplo, na primeira fase, a primeira pergunta

será gerada da tabuada do um, o nível da pergunta cresce de acordo com as respostas certas do jogador, ao acertar três vezes, a tabuada será mudada, no caso da tabuada do um irá para a do dois depois de três acertos, de dois para do três e assim por diante.

4.2 Definição do sistema

4.2.1 Plataforma

O "Tabuada no Espaço" será desenvolvido para sistemas Androids voltados para smartphones e tablets. As justificativas para isso são os números que o mercado móvel apresentou nesses últimos anos, onde houve uma grande hegemonia dos dispositivos Android, com mais de metade da quota de mercado na venda de dispositivos móveis.

Queirós (2013) afirma que a Google Play, loja online de aplicações Android, registrou que mais de 800 mil aplicações e mais de 30 bilhões de downloads foram feitos até o momento do levantamento, esses números abrangem aplicações pagas e gratuitas.

4.2.2 Engine e Linguagem de programação

Para o desenvolvimento e simulação foi escolhido a utilização da engine Corona SDK que utiliza linguagem Lua. Essa engine foi escolhida por aumentar a velocidade e simplificar o desenvolvimento com ajuda de suas bibliotecas. Nations (2016) ressalta que o Corona reduz o tempo de programação drasticamente, usando como o exemplo uma tarefa para ser desenvolvida em Objective-C que gera centenas de linhas de códigos basta usar algumas linhas para realizar em Lua.

Outro benefício que essa engine apresenta é o fato de possuir o próprio controle de seu modelo de memória, dessa forma o desenvolvedor não precisa se preocupar como inicializar ou remover objetos, pois o Corona trata isso automaticamente de acordo com a memória disponível.

4.3 Modelagem do Sistema

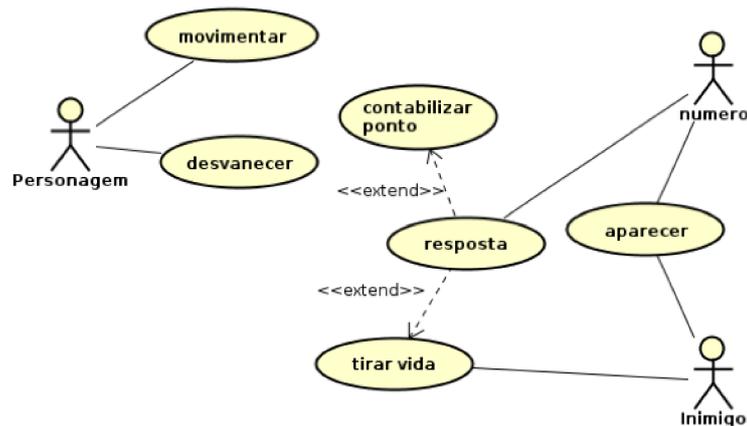
4.3.1 Casos de uso

O modelo de caso de uso, segundo Bezerra (2014), é uma representação das funcionalidades observáveis e dos elementos externos que interagem com ele através da representação com atores, entretanto é possível utilizar os casos de uso em uma perspectiva do "mundo real", isso é, tratar a interação não somente com funcionalidades, mas também com ações que o ator pode fazer.

Os atores no diagrama de casos de uso são representados por bonecos simples feitos de traços, as ações são representadas por caixas arredondadas que descrevem com palavras simples o objetivo daquela ação, as relações entre atores e ações são representadas por traços e a relação entre ações é representada por um linha tracejada com uma seta na ponta para representar que ação está chamando a outra.

A modelagem de caso de uso é parte integrante da especificação de requisitos, moldando os requisitos funcionais do sistema. Na perspectiva dos jogos, pode-se utilizar para visualizar interações entre personagens e objetos para ajudar a definir responsabilidades e ações, conforme a figura 17 mostra.

Figura 17 – Modelagem de casos de uso do Jogo Tabuada no Espaço modelado no Astah Community.



Fonte: do autor, 2017

Os casos de uso abrangem a realidade do sistema, contém o ator personagem que se movimenta no cenário e desvanece após conflitar com um número ou inimigo. Outro personagem é o numeral, o qual é gerado aleatoriamente várias vezes durante a fase, esses números servem para que o jogador responda a questão aritmética, cabe ao número responder ao jogador se a resposta da questão está certa ou errada. O inimigo serve apenas para atrapalhar e tirar chances de acerto caso colida com o personagem.

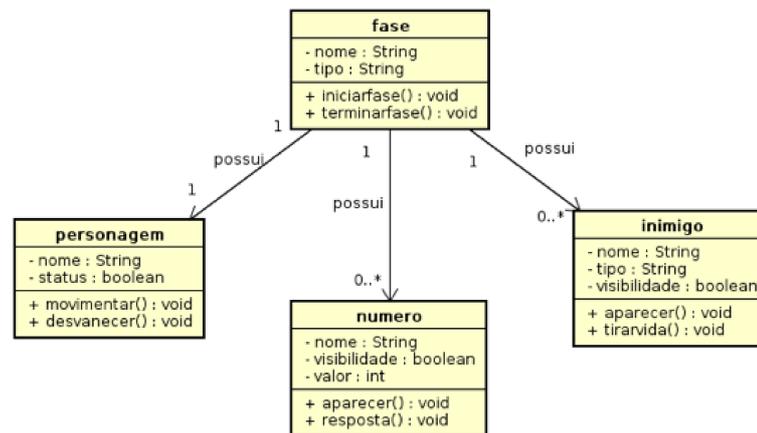
4.3.2 Diagrama de classes

O diagrama de classe, segundo Bezerra (2014), tem como objetivo descrever os vários tipos de objetos no sistema e o relacionamento entre eles. As classes são representadas por retângulos e possuem um nome, métodos e atributos. Esse diagrama pode oferecer três perspectivas, cada uma para um tipo de observador diferente:

- Perspectiva Conceitual: Mostra os conceitos do domínio em estudo;
- Perspectiva de Especificação: Focada nas interfaces de arquitetura;
- Perspectiva de Implantação : Aborda vários detalhes de implementação, tais como navegabilidade, tipo dos atributos, etc.

Foram modelados para esse sistema três classes principais que compõe todo o andamento da fase, cada classe possui sua importância, entretanto a classe fase é a mais importante para o andamento do nível. É possível observar na figura 18 a modelagem de classes que engloba o sistema.

Figura 18 – Diagrama de classes do jogo Tabuada no Espaço modelado no Astah Community.



Fonte: do autor, 2017

A classe fase gerencia o andamento do jogo, ela possui as funções de iniciar a fase e outra de terminar a fase que é chamada sempre que é mudado um status do jogo, isso é, quando o jogador perde uma vida ou responde uma questão certa, o método "terminarfase()" é chamado, e caso o jogador não tenha mais uma vida, o jogo é encerrado. Números são gerados aleatoriamente conforme o andamento da fase, assim como o inimigo é chamado a cada certo tempo de jogo. O jogo pode ser encerrado também caso todas as questões tenham sido respondidas.

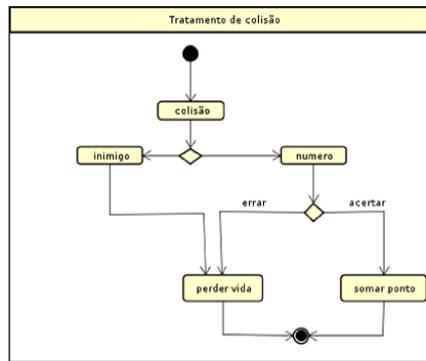
4.3.3 Fluxogramas

Fluxograma, segundo Bezerra (2014) é uma representação de um processo que utiliza símbolos gráficos para representar a natureza e o fluxo do processo. O objetivo é mostrar de forma objetiva o fluxo das informações e elementos, além da sequência que caracteriza o trabalho que está sendo executado. Nessa etapa serão descritos os dois processos mais importantes do jogo: o tratamento de colisão e o fluxo da fase.

4.3.3.1 Tratamento de colisão

A detecção de colisão é fundamental para a maioria dos jogos, para não quebrar a imersão do jogador a colisão deve ser rápida, eficiente e precisa. Para preencher esses requisitos foi feito um diagrama para analisar e determinar o fluxo e reação das colisões. O fluxograma desse processo é mostrado na figura 19.

Figura 19 – Fluxograma da colisão modelado no Astah Community.



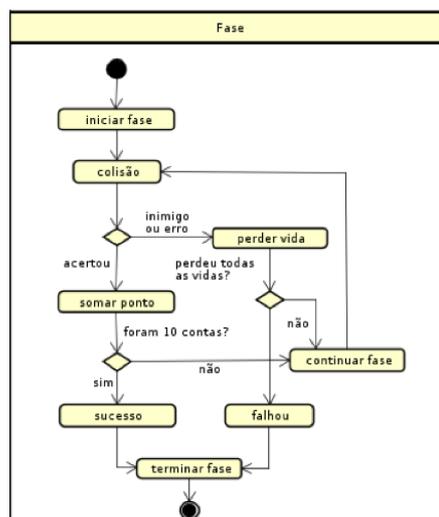
Fonte: do autor, 2017

Quando se trata de um inimigo o fluxo leva a perda de vida. Já o fluxo do número pode ter dois caminhos possíveis, errar a conta, que causa a perda de uma vida, e quando acerta, soma um ponto no placar do jogador. O fluxo daquela colisão termina e outro começa em um próximo impacto entre o personagem e os números.

4.3.3.2 Fluxo da fase

O fluxograma de acontecimentos para a continuidade ou a finalização da fase é o fluxo mais importante do jogo e que determina os acontecimentos da partida. Na figura 20 é possível observar como foi definido a continuidade do processo.

Figura 20 – Fluxograma da fase modelado no Astah Community.



Fonte: do autor, 2017

Após uma colisão é analisado se o jogador irá ganhar um ponto ou perder uma vida,

depois será analisado se aquela foi a última conta da fase ou se o jogador perdeu todas as vidas, caso seja a última ou todas as vidas acabarem, a fase é terminada, em caso negativo, o fluxo da fase retorna para a detecção da próxima colisão.

5 PROTOTIPAGEM DO SISTEMA TABUADA NO ESPAÇO

Prototipação, segundo Pressman e Maxim (2016) é a técnica de desenvolvimento de uma prévia do sistema, essa técnica auxilia os desenvolvedores da aplicação e o usuário, a entenderem o que o sistema realmente precisa, no protótipo são usados telas, imagens e códigos temporários, somente para teste, na maioria dos casos o código não é reaproveitado. O protótipo é usado para que o cliente possa testar o produto e a partir do feedback recebido, se obter uma noção quanto a uma possível necessidade de maior detalhamento dos objetivos do sistema, mudança de características e ajustes de interface.

Com o objetivo de testar a aplicação e verificar se o jogo consegue realizar o objetivo de despertar o interesse do aluno foi desenvolvido um protótipo do jogo "Tabuada no Espaço", os códigos das função principal, do menu e da fase estão nos Apêndices A, B e C respectivamente. Os códigos utilizados não seguem nenhum padrão de desenvolvimento e de reusabilidade sendo feito apenas para o o protótipo do jogo.

5.1 História do jogo

Definir uma história auxilia na imersão do aluno em relação ao jogo, motiva e transfere uma sensação de dever para o jogador, isso torna o jogo mais atrativo. Ambientando no espaço, o "Tabuada no espaço" conta uma curta história de um astronauta, que precisa superar obstáculos e resolver operações aritméticas para passar de nível. O jogo inicia com esse personagem perdendo a comunicação com o seu transporte e sendo jogado para longe de sua nave por culpa de um vírus, sendo colocado assim em uma missão para recuperar sua nave e acabar com o vírus. Foi utilizada uma linguagem fácil para o entendimento da criança, como mostra a figura 21.

Figura 21 – Tela do jogo exibindo história inicial sendo contada através de textos.



Fonte: do autor, 2017

Para colocar no contexto da matemática e um pretexto para serem resolvidos os cálculos aritméticos, o jogo conta para o jogador que a nave não pode partir sem a chave de ignição que está com o passageiro espacial, além do vírus enviar obstáculos na direção do astronauta para capturá-lo, por gostar muito de matemática, acaba oferecendo um desafio extra para o jogador enviando contas de aritmética que precisam ser resolvidas.

5.2 Elementos da tela de jogo

5.2.1 Tela de escolha de níveis

O protótipo do Tabuada no Espaço possui 9 fases divididas em operações de soma, subtração e multiplicação com níveis de dificuldades diferentes. As porcentagens representam o desempenho do jogador em cada fase, sendo que para destravar a próxima fase é preciso que o jogador consiga alcançar o mínimo de 60% de acertos na fase, quando isso não acontece o jogador precisa repetir o mesmo nível novamente. Isso pode ser observado na figura 22.

Figura 22 – Tela do jogo exibindo menu de escolha dos níveis.



Fonte: do autor, 2017

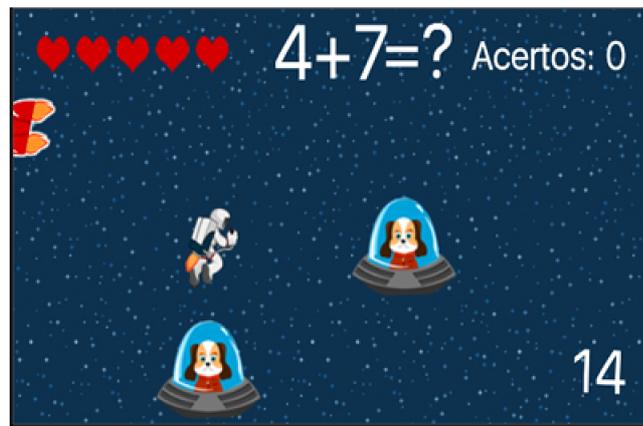
A distribuição dos níveis está feita da seguinte forma:

- Fases de 1 a 3: Fases de soma, cada nível possui 3 tabuadas, o primeiro nível aborda as tabuadas do 1, 2 e 3, o segundo nível a do 4, 5 e 6 e o terceiro nível a do 7, 8 e 9;
- Fases de 4 a 6: Fases de subtração, da mesma forma da primeira fase, cada nível possui 3 tabuadas, o quarto nível aborda as tabuadas de subtração de 1, 2 e 3, o segundo nível do 4, 5 e 6 e o terceiro nível do 7, 8 e 9;
- Fases 7 e 8: São as fases de multiplicação, as tabuadas do 1, 2 e 3 são abordadas no sétimo nível, enquanto a do 4, 5 e 6 na fase 8;
- Fase 9: Fase de divisão, existe apenas uma fase de divisão no jogo, esse nível aborda as tabuadas do 1 ao 3.

5.2.2 Tela de jogo

O jogo apresenta várias informações para o jogador, iniciamos pelos elementos que controlam e informam a quantidade de erros e acertos do jogador, representado por 5 corações, os erros do jogador são contabilizados através de corações perdidos, a fase não é concluída quando o jogador perde todas as suas vidas sem responder todas as questões, outra possibilidade de perder uma vida é acertar os obstáculos. O contador da direita mostra quantas questões foram acertadas desde o começo da fase, pode-se observar esses elementos na figura 23.

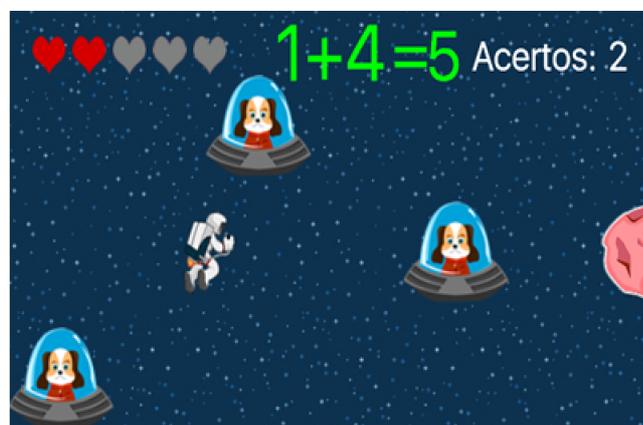
Figura 23 – Tela do jogo exibindo início da fase.



Fonte: do autor, 2017

O terceiro elemento da tela é a conta para ser resolvida, gerada aleatoriamente de acordo com nível no qual se encontra o jogador, esse elemento mostra a resposta certa assim que o jogador passa por um número, no caso de acerto da resposta é mostrada em verde, como mostrado na figura 24.

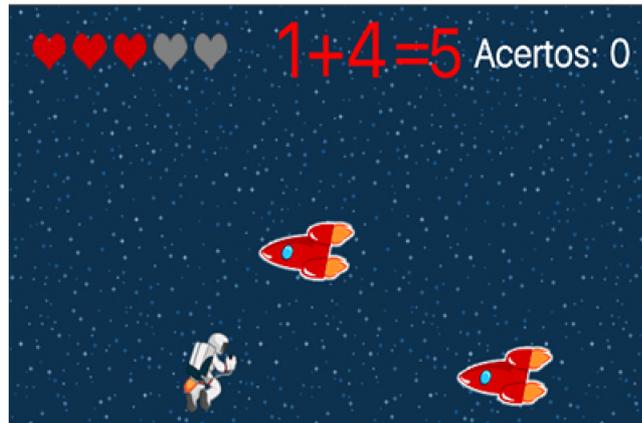
Figura 24 – Tela do jogo exibindo resposta quando a conta está certa.



Fonte: do autor, 2017

Na figura 25, mostra que em caso de erro, o indicador da operação é mostrado em vermelho, mas indica a resposta certa ao jogador, dessa forma o aluno sabe que errou, mas fica ciente de qual seria o resultado correta daquela operação.

Figura 25 – Tela do jogo exibindo feedback quando a resposta não está certa.



Fonte: do autor, 2017

Caso o participante não consiga atingir o percentual de acertos exigido pelo programa, ele poderá recomeçar uma partida quantas vezes desejar, conforme mostrado na figura 26a. Quando o jogador consegue concluir a fase, a mensagem de "Parabéns" será exibida para ele, conforme a figura 26b mostra.

Figura 26 – Telas do jogo exibindo menu final quando (a) fase não foi concluída (b) a fase foi concluída.



Fonte: do autor, 2017

O jogo foi desenvolvido em 2d e dessa forma passa ao usuário a sensação de movimentação horizontal, o personagem desloca da esquerda para direita e os objetos se deslocam

da direita para esquerda, o jogador utiliza a função do touch dos smartphones, para movimentar o personagem para cima e para baixo. Para responder a pergunta, o jogador deve passar por cima do número que corresponde ao resultado da operação aritmética mostrada na parte superior-central da tela, procurando desviar das barreiras que irão surgindo pela frente.

5.3 Teste

Os testes com o protótipo ocorreram com 3 turmas diferentes com a supervisão dos seus respectivos docentes, depois dos testes, as educadoras foram entrevistadas, duas professoras lecionavam para turmas do primeiro ano do ensino fundamental, enquanto a outra ministrava aula para o segundo ano do ensino fundamental, foram aplicados questionários que buscavam analisar os seguintes aspectos:

- Possibilidade de se tornar uma ferramenta de auxílio na aprendizagem;
- Dificuldade de adaptação;
- O aspecto didático;
- A possibilidade de uso nas aulas de matemática.

O questionário aplicado pode ser conferido na íntegra no Anexo A. No primeiro ponto, os resultados foram satisfatórios, todos concordaram que o jogo possui bons argumentos para se tornar uma ferramenta de auxílio na aprendizagem. No aspecto de adaptação os resultados oscilaram, durante as atividades das crianças com o jogo digital, uma das professoras reportou que na fase de domínio da jogabilidade alguns alunos de sua turma tiveram um certa dificuldade para se adaptar o jogo, mas de forma geral foi apontado um grau de adaptação boa dentro do jogo depois dessa fase inicial.

Outro ponto com bons resultados foi o da possibilidade do jogo ser aplicado nas aulas de matemática dos professores entrevistados, todos concordaram e mostraram que tinham interesse de testar aquela ferramenta mais vezes em suas salas de aula.

Apesar de não serem entrevistados diretamente, pode-se observar nas crianças durante o teste que o nível de concentração teve resultados aceitáveis também, durante o teste os jogadores estavam bem concentrados e prestaram bastante atenção nas instruções da fase e nos aspectos do ambiente. Pode se observar também a dificuldade do jogo para as crianças, aspecto em que o resultado foi mediano, apesar de acharem as primeiras fases “fáceis” sentiram bastante dificuldade nas fases de multiplicação e divisão.

6 CONCLUSÃO

Neste trabalho foi proposto o desenvolvimento de um jogo digital para despertar o interesse e engajamento dos alunos no processo de ensino aprendizagem sobre as operações aritméticas nos primeiros anos do Ensino Fundamental. Esta pesquisa foi motivada pelo fato de que o segmento de jogos provoca um interesse muito grande na área da educação, surgindo assim a oportunidade de realizar um projeto que pudesse gerar uma ferramenta para essa área.

Para que o resultado fosse alcançado, foi feito um embasamento teórico onde se buscou na literatura a importância do lúdico no ensino da matemática e dos jogos digitais para educação, procurou-se também durante essa etapa ferramentas que pudessem fornecer os meios necessários para o desenvolvimento deste jogo, onde foi constatado que a criação do jogo utilizando a engine Corona SDK era uma ótima opção por fornecer bibliotecas e ferramentas que tornam o sistema mais estável e oferecem a possibilidade de desenvolvimento multiplataforma.

Durante o levantamento de pesquisas relacionadas na área de jogos educativos, foram escolhidos aqueles que possuíam critérios que estavam relacionados com o objetivo e características deste trabalho. Foi feita uma análise comparativa para constatar que a proposta do jogo "Tabuada no Espaço" possuía características que qualificam seu desenvolvimento.

Uma etapa importante para o desenvolvimento de softwares é o de modelagem e definição do sistema, esse processo torna o sistema mais robusto e facilita modificações em momentos posteriores. Neste trabalho foi utilizado técnicas com base na linguagem UML (casos de uso, fluxogramas e diagramas de classes) muito utilizada em desenvolvimento de sistemas, essa linguagem retrata situações, fluxos e componentes do sistema de uma forma que até uma pessoa sem conhecimento técnico de programação ou de análise de sistemas consiga compreender o que o sistema está propondo. Foi definido durante o processo de modelagem que o sistema possuirá inicialmente 9 fases, divididas entre as tabuadas de soma, subtração, multiplicação e divisão, onde para liberar o próximo nível o jogador terá que acertar 60% das operações aritméticas.

O jogo se propõe a ensinar as tabuadas em um ambiente interativo e imersivo, onde o aluno deve tocar na tela do celular para movimentar o personagem com o objetivo de desviar dos objetos e pegar o número correto para a questão proposta, que é mostrado no canto superior da tela, essa operação se altera depois que o jogador responde a questão. A vantagem de se utilizar um jogo digital educativo é o fato que o erro pode ser minimizado, pois o processo pode ser repetido inúmeras vezes por vontade do aluno. O "Tabuada no Espaço" possui um nível de dificuldade crescente, onde essa evolução ocorre conforme o acerto do aluno, com a repetição acreditasse que o aluno poderá chegar cada vez mais longe dentro do jogo e que pode resultar no aprendizado daquelas operações.

Para verificar se a proposta do jogo poderia realizar a tarefa de estimular e desafiar os alunos nas aulas de matemática foi construído um protótipo do jogo "Tabuada no Espaço" utilizando a

linguagem Lua e a engine Corona SDK para ser jogado em celulares Androids. Os resultados foram levantados através de questionários aplicados aos docentes de três turmas diferentes, sendo duas turmas do primeiro ano do ensino fundamental e uma turma do segundo ano do fundamental. Esses professores tiveram o papel de analisar aspectos como adaptação ao jogo, o aspecto didático e a possibilidade de se tornar uma ferramenta de auxílio na aprendizagem. Os resultados obtidos através destes questionários e o acompanhamento dos testes possibilitam concluir que a utilização do jogo digital proporcionou aos estudantes uma experiência desafiadora e interessante, que deixou as crianças com vontade de ter mais contato com o jogo, dessa forma pode-se afirmar que a proposta deste trabalho foi alcançada.

6.1 Trabalhos Futuros

O trabalho seguirá em desenvolvimento. Existem diversas possibilidades de melhorias para o jogo. Para este primeiro momento, elenca-se as seguintes propostas, a saber:

- Criar perfis e base de dados para gerar relatórios de acertos e erros dos alunos para que o professor responsável possa acompanhar o desenvolvimento da criança;
- Avanço dos testes em escolas para buscar na prática melhorias para o projeto;
- Publicar trabalhos na área de jogos digitais para educação utilizando como tema os resultados obtidos com o "Tabuada no Espaço".

Referências

- ALEXANDRE, C.; SABBATINI, M. **A contribuição dos Jogos Digitais nos processos de aprendizagem**. 2013. Universidade Federal do Pernambuco. Acesso em: 15.10.2016.
- ALMEIDA, R. G. et al. **Desenvolvimento de um Jogo Computacional como Objeto de Aprendizagem para o Ensino da Matemática**. 2014. Universidade Federal Rural de Pernambuco. Acesso em: 02.11.2016.
- ANALYTICS, S. *Android Captured 79 share of global smartphone shipments in 2013*. 2014. Disponível em: <<https://www.strategyanalytics.com/strategy-analytics/blogs/devices/smartphones/smart-phones/2014/01/29/android-captured-79-share-of-global-smartphone-shipments-in-2013>>. Acesso em: 10.12.2016.
- ARANHA, G. **O processo de consolidação dos jogos eletrônicos como instrumento de comunicação e de construção de conhecimento**. Rio de Janeiro, 2004. Acesso em: 7.11.2016.
- BEZERRA, E. *Princípios de análise e projeto de sistemas com UML*. 3. ed. Rio de Janeiro: Elsevier, 2014.
- CLUA, E. W. G.; BITTENCOURT, J. **Desenvolvimento de Jogos 3D: Concepção, Design e Programação**. Rio Grande do Sul, 2005. Acesso em: 22.10.2016.
- CORONA. *Corona SDK by Corona Labs*. 2016. Disponível em: <<https://coronalabs.com/>>. Acesso em: 04.11.2016.
- DAVILA, G. **Uma Introdução no Desenvolvimento Móvel de Jogos com Framework Multi-Plataforma**. São Paulo, 2012. Acesso em: 15.08.2016.
- DELOURA, M. *Game Programming Gems 2*. [S.l.]: Charles River Media, 2001.
- FALKEMBACH, G. A. M. **O Lúdico e os Jogos Educacionais**. Rio Grande do Sul, 2007. Acesso em: 05.01.2017.
- FIORENTINI, D.; LORENZATO, S. *Investigação em educação matemática: percursos teóricos e metodológicos*. São Paulo: Autores Associados, 2006.
- GARTNER. *Gartner Says Five of Top 10 Worldwide Mobile Phone Vendors Increased Sales in Second Quarter of 2016*. 2016. Disponível em: <<http://www.gartner.com/newsroom/id/3415117>>. Acesso em: 05.11.2016.
- HUIZINGA, J. *Homo Ludens*. 4. ed. São Paulo: Editora Perspectiva, 2000.
- KENT, S. L. *The Ultimate History of Video Games: From Pong to Pokémon and Beyond – The Story Behind the Craze that Touched Our Lives and Changed the World*. New York: Three River Press, 2001.
- LORENZATO, S. *Educação infantil e percepção matemáticas*. 3. ed. São Paulo: Autores Associados, 2011.
- LUA. *Linguagem de Programação Lua*. 2017. Disponível em: <<https://www.lua.org/>>. Acesso em: 08.01.2017.

- LUCCHESI, F.; RIBEIRO, B. **Conceituação de Jogos Digitais**. 2012. Universidade Estadual de Campinas. Acesso em: 20.09.2016.
- MACHADO, N. J. *Matemática e Educação: alegorias, tecnologias, jogo, poesia*. 6. ed. São Paulo: Cortez, 2012.
- MATTAR, J. *Games em educação: como os nativos digitais aprendem*. São Paulo: Pearson Prentice Hall, 2010.
- MORAN, J. **Mudando a educação com metodologias ativas**. 2015. Universidade de São Paulo. Acesso em: 18.10.2016.
- NATIONS, D. *How to Get Started Developing iPad Apps*. 2016. Disponível em: <<https://www.lifewire.com/get-started-developing-ipad-apps-1994286>>. Acesso em: 17.12.2016.
- NETO, J. F. B. **Uma Metodologia de Desenvolvimento de Jogos Educativos em Dispositivos Móveis para Ambientes Virtuais de Aprendizagem**. Recife, 2013. Acesso em: 01.11.2016.
- PAIS, L. C. *Ensinar e aprender Matemática*. 2. ed. Belo Horizonte: Autêntica editora, 2013.
- PEREIRA, L. C. O.; SILVA, M. L. *Android para desenvolvedores*. Rio de Janeiro: Brasport, 2009.
- PRESSMAN, R. S.; MAXIM, B. R. *Engenharia de software: uma abordagem profissional*. 8. ed. Porto Alegre: AMGH, 2016.
- SCHNEIDER, C. L. *Matemática: O Processo de Ensino-Aprendizagem*. 2007. Disponível em: <<http://www.somatematica.com.br/artigos/a32/>>. Acesso em: 10.12.2016.
- SILVA, S. S. R.; NOBREGA, S. M.; JACOB, A. F. L. **Labirinto do Rato: jogo educacional infantil para dispositivos móveis**. 2011. Universidade da Amazônia. Acesso em: 01.11.2016.
- SMOLE, K.; DINIZ, M. I.; CANDIDO, P. *Jogos de matemática de 1 ao 5 ano*. Porto Alegre: Artmed, 2007.

Apêndices

APÊNDICE A – CÓDIGO DO MAIN.LUA

```
--main.lua
--Esconder a barra de status do dispositivo
display.setStatusBar( display.HiddenStatusBar )

--Importando bibliotecas--
local composer = require "composer"
local niveis = "niveis.json"
local loadsave = require( "loadsave" )

--Criar arquivo niveis.json no dispositivo novo
if loadsave.loadTable("niveis.json") == nil then

niveis = {

nive11 = 0;
nive12 = 0;
nive13 = 0;
nive14 = 0;
nive15 = 0;
nive16 = 0;
nive17 = 0;
nive18 = 0;
nive19 = 0;

}

loadsave.saveTable( niveis, "niveis.json" )

end

--Criar menu principal
composer.gotoScene( "menu" )
```

APÊNDICE B – CÓDIGO DO MENU.LUA

```

-- menu.lua
--- Importando bibliotecas ---
local composer = require( "composer" )
local json = require( "json" )
local loadsave = require( "loadsave" )
local carregarSons = require("sons")

--Criando cena--
local scene = composer.newScene()

--Importando audio
local menuprincipal = audio.loadStream( "theme.mp3" )

--Objeto botao de iniciar--
local playBtn

--Carregando dados do jogador--
local loadedinfo = loadsave.loadTable( "saveSettings.json" )

--VARIABLES GERAIS--
aux = 0;

local function comecarJogo()

composer.gotoScene( "escolhafase", "fade", 600 )
end

function scene:create( event )

local sceneGroup = self.view

local background = display.newImageRect( "matematica.png", 491, 273 )
background.anchorX = 0
background.anchorY = 0
background.x, background.y = -9, 23

playBtn = display.newText( "JOGAR", 10, 10 ,native.font, 25 )

playBtn.x = display.contentWidth*0.5 + 15
playBtn.y = display.contentHeight - 106

playBtn.addEventListener("touch", comecarJogo)
playBtn.alpha = 0.01;

sceneGroup:insert( background )
sceneGroup:insert( playBtn )

end

```

APÊNDICE C – CÓDIGO DO LEVEL.LUA

```

--level geral
-- Importação de bibliotecas
local composer = require( "composer" )
local scene = composer.newScene()
local json = require( "json" )
local loadsave = require( "loadsave" )
local menu = require("menu")
local saveSettings = "tabelaresultados.json"
local niveis = "niveis.json"
local sons = require("sons");
local physics = require "physics"
--Importação de arquivo com as informações do jogador
local loadedinfo = loadsave.loadTable( "saveSettings.json" )
local loadniveis = loadsave.loadTable( "niveis.json" )
--requisitos do sistema
system.activate( "multitouch" )
--Declarações de sons
local lasersound = audio.loadSound( "laser.mp3" )
local cartaplay = audio.loadSound( "cartacapturada.mp3" )
local escudomusic = audio.loadSound( "escudoplay.mp3" )
local clicksound = audio.loadSound("click.m4a")
local danosound = audio.loadSound( "danosound.mp3" )
local errosound = audio.loadSound( "sad.mp3" )
local acertosound = audio.loadSound( "happy.mp3" )
local coisasboassound = audio.loadSound( "positivo.mp3" )
local aceleradorplay = audio.loadSound( "acelerar.mp3" )

-- VARIAVEIS GERAIS
local screenW, screenH, halfW = display.contentWidth,
display.contentHeight, display.contentWidth*0.5
local vpause = 0; mudancasdefundo = 0;
local LEFT = 100 -- TOP
local CENTER = display.contentCenterY + 20 -- CENTRO
local RIGHT = display.contentHeight - 60 -- BAIXO

--DECLARACOES DE FUNCOES--
--Funcao para tornar o fundo ciclico
function moverFundo(event)
    if(vpause == 0) then
        local tDelta = event.time - tPrevious
        tPrevious = event.time

        local xOffset = ( velocidade * tDelta )

        fundo.x = fundo.x - xOffset
        fundo2.x = fundo2.x - xOffset

        if (fundo.x + fundo.contentWidth) < 0 then
            fundo:translate( 637 * 2, 0)
        end
        if (fundo2.x + fundo2.contentWidth) < 0 then
            fundo2:translate( 637 * 2, 0)
        end
    end
end

```

```

        end
    end
end
---INICIO DO CENARIO---
function scene:create( event )
    -- Inicia a física do jogo
    physics.start();
    --Cria grupo da cena
    local sceneGroup = self.view
    --Definicao do plano de fundo em movimento
    local fundo = display.newImageRect( "plano-fundo.png" , 637, 425 ) --fundo-spaceverde2

    fundo.anchorY = 0;
    fundo.anchorX = 0;
    fundo.x = 0
    fundo.y = 0;
    --fundo.yScale = 1.1;
    --local fundo2 = display.newImage( "fundo2000.png" ,display.contentWidth,
                                        display.contentHeight, yes)
    local fundo2 = display.newImageRect( "plano-fundo.png" , 637, 425 ) --fundo-spaceverde2
    fundo2.anchorY = 0;
    fundo2.anchorX = 0;
    fundo2.x = 637;
    fundo2.y = 0;
    --- VELOCIDADE DAS COISAS ----
    velocidade = 0.12;velocidadedascoisas = -120; delaycartas = 35;
    mudanca = 0;mudancatipo = 0;mudancafade = 0;gametime = 0;
    local tPrevious = system.getTimer()
    --Insere fundos ao grupo da cena
    sceneGroup:insert( fundo)
    sceneGroup:insert( fundo2)
    --Adiciona a função moverFundo para ser chamada constantemente durante o jogo
    Runtime.addEventListener( "enterFrame", moverFundo );
    --variaveis
    nscore = 0; tempodejogo = 0;
    -----
    -- crinado o Sprite que vai se mover
    local bobSpriteOptions =
    {
        width = 50,
        height = 52,
        numFrames = 6
    }
    local bobSheets = graphics.newImageSheet( "bob2.png", bobSpriteOptions)
    --Tabela de sequencias
    local sequences_bob = {
        {
            name = "normalRun",
            start = 1,
            count = 6,
            time = 900,
            loopCount = 0,
            loopDirection = "forward"
        };
        {
            name = "pause",

```

```

        start = 1,
        count = 1,
    };
}
--Declarando o personagem principal
local bob = display.newSprite( bobSheets, sequences_bob )
bob.x = 150;
bob.y = CENTER;
--Movimentacao do personagem
local moveon = 1;
function moveonoff()
    moveon = 1;
end
local function movimentarPersonagem( event )
    if (vpause==0) then
        if ( event.phase == "moved" ) then
            local dY = event.y - event.yStart
            if(moveon == 1) then
                if ( dY > 10 ) then
                    --swipe right
                    local spot = RIGHT
                    moveon = 0;
                    if ( bob.y == LEFT ) then
                        spot = CENTER
                    end
                    transition.to( bob, { time=120, y=spot, onComplete=moveonoff } )
                elseif ( dY < -10 ) then
                    --swipe left
                    local spot = LEFT
                    moveon = 0;
                    if ( bob.y == RIGHT ) then
                        spot = CENTER
                    end
                    transition.to( bob, { time=120, y=spot, onComplete=moveonoff } )
                end
            end
        end
    end
end
return true
end
Runtime:addEventListener( "touch", movimentarPersonagem )
--iniciar a movimentação do personagem
bob:play();
--texto de avisos
local avisos = display.newText( "", 324, 54, native.font, 44 )
avisos:setFillColor( 1, 0, 0)
avisos.isVisible = false;
--Declaracao de coracao
local coracaoSpriteOptions =
{
    width = 25,
    height = 25,
    numFrames = 2
}
local coracaoSheets = graphics.newImageSheet("coracao-play2.png", coracaoSpriteOptions)
-- sequences table

```

```
local sequence_coracao = {
-- consecutive frames sequence
{
    name = "1",
    start = 1,
    count = 1,
    loopCount = 0,
    loopDirection = "forward"
};

{
    name = "2",
    start = 2,
    count = 2,
    loopCount = 0,
    loopDirection = "forward"
};
}
local coracao = display.newSprite( coracaoSheets, sequence_coracao )
coracao:setSequence("1");
coracao.x = 30;
coracao.y = 55;
coracao:play();
local coracao2 = display.newSprite( coracaoSheets, sequence_coracao )
coracao2:setSequence("1");
coracao2.x = 60;
coracao2.y = 55;
coracao2:play();
local coracao3 = display.newSprite( coracaoSheets, sequence_coracao )
coracao3:setSequence("1");
coracao3.x = 90;
coracao3.y = 55;
coracao3:play();
local coracao4 = display.newSprite( coracaoSheets, sequence_coracao )
coracao4:setSequence("1");
coracao4.x = 120;
coracao4.y = 55;
coracao4:play();
local coracao5 = display.newSprite( coracaoSheets, sequence_coracao )
coracao5:setSequence("1");
coracao5.x = 150;
coracao5.y = 55;
coracao5:play();
--inserindo coracoes na cena
sceneGroup:insert(coracao);
sceneGroup:insert(coracao2);
sceneGroup:insert(coracao3);
sceneGroup:insert(coracao4);
sceneGroup:insert(coracao5);
--inserindo personagem principal na cena
sceneGroup:insert(bob)
--variaveis
local contjump = 0; contimpulso =0;
--textos de aviso
local textoprincipal = display.newText( "?", 306, 51, native.font , 50 )
sceneGroup:insert(textoprincipal)
```

```
local acerto = 0;tipodeconta=0;

naopegarnumero = 0; vetor = 1; historicoaux = {};

subirdificuldade = 0; valor1 = 1;

local function calculo()

    if subirdificuldade == 1 then
        valor1 = valor1+1;
        subirdificuldade = 0;
    end

    --math.random (0,5 )
    valor2 = math.random (0,9 )
    textoprincipal.text = "=?"

    textoprincipal:setFillColor(255,255,255);

    if(tipodeconta == 0) then
        textocalculo = display.newText( valor1.."+"..valor2, 240, 52, native.font, 50 )
        acerto = valor1+valor2;
    elseif(tipodeconta==1) then

        if(valor1<valor2)then
            valor2 = math.random(0,valor1)
        end

        textocalculo = display.newText( valor1.."-"..valor2, 240, 52, native.font, 50 )
        acerto = valor1-valor2;

    end

    sceneGroup:insert(textocalculo)
end

acertostxt = display.newText("Acertos:", 393, 54, native.font, 26 )
acertosnum = display.newText( "0", 455, 54, native.font, 26 )
acertosint = 0;

sceneGroup:insert(acertostxt);
sceneGroup:insert(acertosnum);

calculo();

coracaoacerto = 5;

local pesquerda = display.newRect( 0, 0, 0, display.contentHeight ); -- PAREDE ESQUERDA
pesquerda.anchorX = 0;
pesquerda.anchorY = 0;
pesquerda.x = - 10;
pesquerda.type = "parede";
physics.addBody( pesquerda, "static")

local pesquerda = display.newRect( 0, 0, 0, display.contentHeight ); -- PAREDE ESQUERDA
```

```
pesquerda.anchorX = 0;
pesquerda.anchorY = 0;
pesquerda.x = - 10;
pesquerda.type = "parededoosso";
physics.addBody( pesquerda, "static")

local pedireita = display.newRect( display.contentWidth, 0, 0, display.contentHeight );
pedireita.anchorX = 0;
pedireita.anchorY = 0;
pedireita.type = "parededoosso"
physics.addBody( pedireita, "static")

local pcima = display.newRect( 0, 0, display.contentWidth, 0 );
pcima.anchorX = 0;
pcima.anchorY = 0;
physics.addBody( pcima, "static")

mpause = display.newGroup()
menupause = display.newImageRect("menuzao.png", 350, 256)
menupause.x = display.contentWidth * 0.5;
menupause.y = display.contentHeight * 0.5;
mpause:insert(menupause)

local fuelmenu = display.newText( "0", 111, 159, native.font, 12 )
local shieldmenu = display.newText( "0", 121, 187, native.font, 12 )
local scoremenu = display.newText( "0", 120, 217, native.font, 12 )
local lettersmenu = display.newText( "0", 125,249, native.font, 12 )

menusoundcaixa = display.newRect(120, 75, 80, 20 )
menusoundcaixa:setFillColor(black)
menusoundcaixa.alpha = 0.0
menusound = display.newText("Sound: ON", 120, 75, native.newFont( "Comic Sans Ms"), 15);

local function soundonoff()
    if(audio.getVolume()==0) then
        audio.setVolume(1)
        menusound.text = "Sound: ON"
    elseif(audio.getVolume()==1) then
        audio.setVolume(0)
        menusound.text = "Sound:OFF"
    end
end

end

menusound:addEventListener( "tap", soundonoff )
mpause:insert(menusoundcaixa)
mpause:insert(menusound)
mpause:insert(fuelmenu)
mpause:insert(shieldmenu)
mpause:insert(scoremenu)
mpause:insert(lettersmenu)

local returnbtn =display.newRect(70, 300, 156, 48 )
returnbtn.x = menupause.x + 90;
returnbtn.y = menupause.y - 20;

local configbtn = display.newRect(70, 3000, 156, 48 )
configbtn.x = returnbtn.x ;
```

```
configbtn.y = returnbtn.y + 55 ;

local exitbtn = display.newRect(70, 300, 156, 48 )
exitbtn.x = returnbtn.x;
exitbtn.y = returnbtn.y + 110;
returnbtn.alpha = 0.01;
configbtn.alpha = 0.01;
exitbtn.alpha = 0.01;

local menuescolha = 0;
mpause:insert(returnbtn);
mpause:insert(configbtn);
mpause:insert(exitbtn);
mpause.isVisible = false;

local options = {
    isModal = true,
    effect = "fade",
    time = 400,
    params = {
        sampleVar = "my sample variable"
    }
}

local function exit()
    local clickplay = audio.play( clicksound)
    if(menuescolha == 0) then
        nscore = 0;
        ndecartas = 0;
        --physics.pause( );

        --if carta ~= nil then
        --carta:removeEventListener( "collision", carta )
        --end

        exitbtn:removeEventListener( "tap", exit )
        returnbtn:removeEventListener( "tap", resume )
        configbtn:removeEventListener( "tap", howtoplaybtn )
        pausebtn:removeEventListener( "tap", pausefunction )
        sceneGroup:removeSelf( );

        composer.gotoScene( "menu", "fade", 400 )
        composer.removeScene( "level1" )
    elseif menuescolha == 1 then

        local function saida( event )
            if event.action == "clicked" then
                local i = event.index
                if i == 1 then
                    menuescolha =0;
                    exit()
                elseif i == 2 then
                    end
                end
            end
        end
    end
end
```

```
        local alert = native.showAlert( "CONFIRMACAO",
                                        "Voce tem certeza que deseja sair",
                                        { "SIM" , "CANCELAR"},
                                        saida )
    end
end

local function terminarjogo()

    audio.pause()
    vpause = 1;
    pauseframe();
    physics.pause( );
    pausebtn.isVisible = false;

    local function gerarmenufinal()
        if(acertosint < 6) then
            caixaMENU = display.newImageRect( "gameover.png" , 250, 150 )
        elseif acertosint > 5 then
            caixaMENU = display.newImageRect( "gameover-bom.png" , 250, 150 )
        end
        caixaMENU.x = display.contentWidth * 0.5; caixaMENU.y = display.contentHeight * 0.5 + 2
        caixaMENU.alpha = 0.70;

        txtbackMENU = display.newText( "Voltar ao menu",
                                       display.contentWidth * 0.5,
                                       display.contentHeight * 0.5+60,
                                       native.systemFont,
                                       20)

        txtbackMENU.alpha = 0.01;
        txtplayagain = display.newText( "Próximo Nivel",
                                       display.contentWidth * 0.5,
                                       display.contentHeight * 0.5+15,
                                       native.systemFont,
                                       20)

        txtplayagain.alpha = 0.01;

        aux = loadniveis.nivel1;

        loadniveis.nivel1 = acertosint * 10;
        acertostexto = display.newText(loadniveis.nivel1.."%", 277, 155, native.systemFont, 20)
        acertostexto:setFillColor(black);

        if(loadniveis.nivel1 < aux ) then

            loadniveis.nivel1 = aux;

        end

        niveis = {

            nivel1 = loadniveis.nivel1;
            nivel2 = loadniveis.nivel2;
            nivel3 = loadniveis.nivel3;
            nivel4 = loadniveis.nivel4;
```

```
nivel5 = loadniveis.nivel5;
nivel6 = loadniveis.nivel6;
nivel7 = loadniveis.nivel7;
nivel8 = loadniveis.nivel8;
nivel9 = loadniveis.nivel9;

}

loadsave.saveTable( niveis, "niveis.json" )

save_settings()

sceneGroup:insert(caixamenu)
sceneGroup:insert(txtbackmenu)
sceneGroup:insert(acertostexto)
sceneGroup:insert(txtplayagain)
txtplayagain:addEventListener( "touch", playagain )
txtbackmenu:addEventListener( "touch", exit )
end

if(nscore > loadedinfo.valorscore) then
    gerarmenufinal();
elseif loadedinfo.valorscore > nscore then
    gerarmenufinal();
end
end
end

numerodoacerto = 0;

local function trocarcoracoes()

    if(coracaoacerto == 5) then
        coracao5:setSequence("2");
        coracaoacerto = 4;
    elseif(coracaoacerto == 4) then
        coracao4:setSequence("2");
        coracaoacerto = 3;
    elseif(coracaoacerto == 3) then
        coracao3:setSequence("2");
        coracaoacerto = 2;
    elseif(coracaoacerto == 2) then
        coracao2:setSequence("2");
        coracaoacerto = 1;
    elseif(coracaoacerto == 1) then
        coracao:setSequence("2");
        terminarjogo();
    end
end
end
end

auxacerto = 0;

local function acertarconta(event)
    local a = event.target
```

```
numeroacerto = event.target.numero;

if( naopegarnumero == 0 ) then
  if(event.target.numero == acerto) then
    boasosoundplay = audio.play( acertosound)

    avisos.text = acerto;
    testedeenvio(1);
    textoprincipal.text = "= "
    avisos.isVisible = true;
    avisos.setFillColor(0,255,0);
    textoprincipal:setFillColor(0,255,0);
    textocalculo:setFillColor(0,255,0);
    a:removeSelf()
    naopegarnumero = 1;
    acertosint = acertosint+1;
    acertosnum.text = tostring(acertosint);
    trocarcoracaoes()

    if(auxacerto ==0) then
      auxacerto = auxacerto + 1;
    elseif auxacerto == 1 then
      auxacerto = auxacerto + 1;
    elseif auxacerto == 2 then
      subirdificuldade = 1;
      auxacerto =0;
    end
  end

else
  danosoundplay = audio.play(erroresound);
  historico[vetor][1] = loadedinfo.aspirador;
  historico[vetor][2] = textocalculo.text ;
  historico[vetor][3] = 0;
  avisos.text = acerto;
  textoprincipal.text = "= "
  avisos.isVisible = true;
  testedeenvio(0); -- atencao para função
  avisos.setFillColor(255,0,0);
  textoprincipal:setFillColor(255,0,0);
  textocalculo:setFillColor(255,0,0);
  a:removeSelf();
  naopegarnumero = 1;
  trocarcoracaoes()
end

  if(vetor==10) then
    ganhar = 1;
    terminarjogo()
  end

end

end
```

```
iniciodocenario = 300;
local grupocenario = display.newGroup( );
local cenariopos = 0;

local function criadordecenario()
    if vpause == 0 then

        if(cenariopos == 0) then
            local parede2 = display.newImage( "nave.png");
            parede2.x = 240 + iniciodocenario;
            parede2.y = 30;
            physics.addBody( parede2,"kinematic",{friction=0, bounce=0})
            grupocenario:insert(parede2)

            parede2:setLinearVelocity( velocidadedascoisas, 0 )
        elseif(cenariopos==1) then
            local parede5 = display.newImageRect( "superhomemtemp.png", 200, 50);
            parede5.x = 240 + iniciodocenario;
            parede5.y = 175;
            physics.addBody( parede5, "kinematic",{friction=0, bounce=0})
            grupocenario:insert(parede5)
            parede5:setLinearVelocity( -50, 0 )
            parede5.type = "nave"
            parede5.collision = endgame
            parede5:addEventListener( "collision", parede5 )
        end

        posicaoocenario = math.random(0,2); gerarposicao = 0;

        local function posicionarocenario()
            if(posicaoocenario == 0) then
                gerarposicao = 100;
                posicaoocenario = math.random(1,2)
            elseif posicaoocenario ==1 then
                gerarposicao = 180;
                posicaoocenario = math.random(0,2)
            --elseif posicaoocenario == 2 then
            --gerarposicao = 210;
            elseif posicaoocenario == 2 then
                gerarposicao = 260;
                posicaoocenario = math.random(0,1)
            end
        end

        local function planeta()
            posicionarocenario();
            dog = display.newImageRect( "planeta05.png", 80, 65);
            dog.rotation = 2;
            dog.x = 490; dog.y = gerarposicao;
            dog.type = "nave"
            local naveshape = { 0,-33, 20,-24, 35,28, 20,21, -32,32,-35,30, -15,-20 }
            physics.addBody(dog, "kinematic", {shape = naveshape});
            dog:setLinearVelocity( velocidadedascoisas-10, 0 )
            dog.collision = endgame
        end
    end
end
```

```
dog:addEventListener( "collision", dog )
grupocenario:insert(dog)
end

local function nave()
    posicionarcenario();
    nave = display.newImageRect( "nave.png", 80, 65);
    nave.rotation = 2;
    nave.x = 490; nave.y = gerarposicao;
    nave.type = "nave"
    local naveshape = { 0,-33, 20,-24, 35,28, 20,21, -32,32,-35,30, -15,-20 }
    physics.addBody(nave, "kinematic", {shape = naveshape});
    nave:setLinearVelocity( velocidadedascoisas-10, 0 )
    nave.collision = endgame
    nave:addEventListener( "collision", nave )
    grupocenario:insert(nave)

end

local contagem = 0;

local function numeros()
    posicionarcenario();

    conta = math.random(0,20)

    if(conta ~= acerto)then
        contagem = contagem + 1;

        if contagem == 5 then
            conta = acerto;
            contagem = 0;
        end
    end

end

local function satelitef()
    satelite = display.newImageRect( "foginete.png", 70, 70)
    aleatorio = math.random( 150,300 )
    satelite.x = -10; satelite.y = aleatorio;
    satelite.type = "nave"
    local naveshape = { 0,-33, 20,-24, 35,28, 20,21, -32,32,-35,30, -15,-20 }
    physics.addBody(satelite, {shape = naveshape});
    satelite:setLinearVelocity(-velocidadedascoisas - 10 , velocidadedascoisas+80 )
    --satelite:setLinearVelocity( 0, velocidadedascoisas )
    satelite.isSensor = true;
    satelite.gravityScale = 0;
    satelite.collision = endgame
    satelite:addEventListener( "collision", satelite )
    grupocenario:insert(satelite)

end

local function cenariocriaraleatorio()
```

```
if(vpause ==0) then
  if(cenarioaleatoriocriado == 1) then
    cenariosaleatorios = math.random(0,500)
    if(cenariosaleatorios >= 0 and cenariosaleatorios <= 100 ) then
      numeros();
    elseif cenariosaleatorios > 100 and cenariosaleatorios <= 200 then
      planeta();
    elseif cenariosaleatorios > 200 and cenariosaleatorios <= 300 then
      satelitef2()
    elseif cenariosaleatorios > 300 and cenariosaleatorios <= 450 then
      nave()
    elseif cenariosaleatorios > 450 and cenariosaleatorios <= 500 then
      numeros();
    end
  end
end

sceneGroup:insert(grupocenario)
timer.performWithDelay( 1100, cenariocriaraleatorio, 0 )

sceneGroup:insert(mpause)

end

end

Runtime:addEventListener( "system", onSystemEvent )

end -- final da funcao criar scene

function scene:show( event )
  local sceneGroup = self.view
  local phase = event.phase

  if phase == "will" then

  elseif phase == "did" then
    musicambienteplay = audio.play( musicambiente, {channel =1, loops=-1} )
  end
end

function scene:hide( event )
  local sceneGroup = self.view
  local phase = event.phase

  if event.phase == "will" then
    vpause = 1;
  elseif phase == "did" then
  end
end

function scene:destroy( event )

  local sceneGroup = self.view
  physics.stop();
```

```
audio.stop();
scene.evento = "";
package.loaded[physics] = nil
physics = nil
--ads.hide();
audio.dispose(cartaplay)
audio.dispose(escudomusic)
audio.dispose(bossmusic)
audio.dispose(danobossmusic)
audio.dispose(clicksound)
audio.dispose(danosound)
audio.dispose(coisasboassound)
audio.dispose(sugadorsound)
audio.dispose(aceleradorplay)

cartaplay = nil;
escudomusic = nil;
bossmusic = nil;
danobossmusic = nil;
clicksound = nil;
danosound = nil;
coisasboassound = nil;
sugadorsound = nil;
aceleradorplay = nil;

end

scene:addEventListener( "create", scene )
scene:addEventListener( "show", scene )
scene:addEventListener( "hide", scene )
scene:addEventListener( "destroy", scene )

return scene
```

Anexos

ANEXO A – QUESTIONÁRIO APLICADO AOS PROFESSORES

Anexo – Entrevista com os(as) professores(as)

Para preenchimento do(a) professor(a)

Nome (opcional)

Curso de formação Tempo de atuação

Nome da instituição de ensino

Endereço da instituição

Série/ano que leciona Quantidade de alunos por turma

Município Estado

Questionário

1. Você participou nos últimos 2 (dois) anos de cursos de aperfeiçoamento (formação continuada relacionados ao ensino da Matemática)?
() Sim () Não
2. Nas suas formações houve a utilização de jogos para o ensino da Matemática?
() Sim () Não
3. Você julga importante utilizar a atividade lúdica para auxiliar no processo de ensino e aprendizagem de matemática?
() Discordo completamente
() Discordo parcialmente
() Indiferente
() Concordo parcialmente
() Concordo completamente
4. Você já havia utilizado jogos digitais educativos com os seus alunos?
() Discordo completamente
() Discordo parcialmente
() Indiferente
() Concordo parcialmente
() Concordo completamente
5. As crianças tiveram dificuldades para manipular o jogo Tabuada no Espaço?
() Discordo completamente
() Discordo parcialmente
() Indiferente
() Concordo parcialmente
() Concordo completamente

6. Quando os alunos erravam aparecia na tela o resultado da operação na tela (feedback), a resposta ajudou na performance das próximas partidas?
-) Discordo completamente
) Discordo parcialmente
) Indiferente
) Concordo parcialmente
) Concordo completamente
7. As crianças melhoram o desempenho com as operações aritméticas realizadas na sala após o uso do Tabuada no Espaço?
-) Discordo completamente
) Discordo parcialmente
) Indiferente
) Concordo parcialmente
) Concordo completamente
8. O interesse em estudar matemática aumentou após a experiência com o jogo digital educativo na sala?
-) Discordo completamente
) Discordo parcialmente
) Indiferente
) Concordo parcialmente
) Concordo completamente
9. Você acredita que o jogo digital educativo pode potencializar o processo de ensino e aprendizagem de matemática?
-) Discordo completamente
) Discordo parcialmente
) Indiferente
) Concordo parcialmente
) Concordo completamente
10. Você admite que o jogo digital educativo pode potencializar o processo de ensino e aprendizagem de matemática?
-) Discordo completamente
) Discordo parcialmente
) Indiferente
) Concordo parcialmente
) Concordo completamente
11. Você ficou satisfeito(a) com os resultados obtidos a partir do manuseio do jogo Tabuada no Espaço?
-) Discordo completamente
) Discordo parcialmente
) Indiferente
) Concordo parcialmente
) Concordo completamente
12. Você recomendaria o uso do Tabuada no Espaço nas aulas de matemática?
-) Discordo completamente
) Discordo parcialmente
) Indiferente
) Concordo parcialmente
) Concordo completamente