

UNIVERSIDADE FEDERAL DE UBERLÂNDIA
FACULDADE DE ENGENHARIA ELÉTRICA – FEELT
NÚCLEO DE INTELIGÊNCIA ARTIFICIAL

Carlos Anibal Vieira Araújo Júnior

**Projeto e Implementação de um Montador Mobile de Autoria
para Conteúdo Multimídia, Multiplataforma e Multidevice
para M-Learning Utilizando a Plataforma Corona SDK
Aderente aos Processos Humanos de Cognição**

UBERLÂNDIA – MG

2017

Dados Internacionais de Catalogação na Publicação (CIP)
Sistema de Bibliotecas da UFU, MG, Brasil.

A663p
2017

Araújo Júnior, Carlos Anibal Vieira, 1986-

Projeto e implementação de um montador mobile de autoria para conteúdo multimídia, multiplataforma e multidevice para M-Learning utilizando a plataforma corona SDK aderente aos processos humanos de cognição / Carlos Anibal Vieira Araujo Junior. - 2017.

68 f. : il.

Orientador: Luciano Vieira Lima.

Dissertação (mestrado) - Universidade Federal de Uberlândia,
Programa de Pós-Graduação em Engenharia Elétrica.

Inclui bibliografia.

1. Engenharia elétrica - Teses. 2. Sistemas de comunicação móvel. - Teses. 3. Ensino a distância - Teses. 4. Inteligencia artificial - Aplicações educacionais. - Teses. I. Lima, Luciano Vieira, 1960-. II. Universidade Federal de Uberlândia. Programa de Pós-Graduação em Engenharia Elétrica. III. Título.
-

Carlos Anibal Vieira Araújo Júnior

**PROJETO E IMPLEMENTAÇÃO DE UM MONTADOR MOBILE DE
AUTORIA PARA CONTEÚDO MULTIMÍDIA, MULTIPLATAFORMA
E MULTIDEVICE PARA M-LEARNING UTILIZANDO A
PLATAFORMA CORONA SDK ADERENTE AOS PROCESSOS
HUMANOS DE COGNIÇÃO**

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Elétrica – Mestrado, da Faculdade de Engenharia Elétrica da Universidade Federal de Uberlândia, como requisito parcial à obtenção do Título de Mestre em Engenharia Elétrica.

Área de Concentração: Inteligência Artificial

Orientador: Luciano Vieira Lima

UBERLÂNDIA – MG

2017

CARLOS ANIBAL VIEIRA ARAÚJO JÚNIOR

**PROJETO E IMPLEMENTAÇÃO DE UM MONTADOR MOBILE DE AUTORIA
PARA CONTEÚDO MULTIMÍDIA, MULTIPLATAFORMA E MULTIDEVICE
PARA M-LEARNING UTILIZANDO A PLATAFORMA CORONA SDK ADERENTE
AOS PROCESSOS HUMANOS DE COGNIÇÃO**

Dissertação final, apresentada à Universidade Federal de Uberlândia, como requisito parcial à obtenção do título de Mestre em Engenharia Elétrica.

Banca Examinadora:

Prof. Dr. Luciano Vieira Lima (orientador)
Universidade Federal de Uberlândia

Prof. Dr. Luciano Martins Neto
Universidade Federal de Uberlândia

Prof. Dr. Reny Cury Filho
PMU

Uberlândia, 13 de julho de 2017.

DEDICATÓRIA

Dedico esse trabalho a Deus, a minha amada esposa, aos meus pais e irmãos pelo apoio incondicional, ao meu tio por ter me incentivado a seguir este caminho e ao meu filho, por já estar me motivando a realizar grandes feitos.

AGRADECIMENTOS

A Deus, por estar continuamente ao meu lado durante esta caminhada.

Aos meus amados pais, Lucília e Carlos, pelo amor, apoio, carinho e inestimável ajuda que me proporcionaram.

Agradeço também aos meus queridos irmãos, Thaís, Leandro e Lucas, pela torcida incondicional e pelos momentos de alegria, e as minhas sobrinhas, Nathália e Maria Lucília, que vibram a cada conquista do titio.

À minha família, que contribuíram com conversas, reflexões e pensamentos. Obrigada pelo apoio e carinho.

A minha esposa, Lorrana, por ser quem esteve sempre ao meu lado neste tempo de estudo, me ajudando na realização desta conquista e que em todos os momentos me forneceu, carinho, motivação e conforto para cada obstáculo que encontrei.

Ao meu orientador, Luciano, pelo apoio, ensinamentos e ter me incentivado a seguir este caminho e estado presente sempre que necessitei de sua ajuda.

Aos membros da banca examinadora, pela dedicação em avaliar este trabalho e pelas contribuições e considerações.

RESUMO

O uso crescente de tecnologia mobile, antes pela geração Y, hoje pelos nativos digitais, também pelo público da terceira idade, vem abrindo uma lacuna, uma distância entre professores e alunos quanto ao material didático disponibilizado de forma impressa e até mesmo na nuvem. Os e-books, principalmente para em plataformas de ensino/aprendizagem em AVAs, EAD, nos formatos .epub, .mobi e .pdf, cujos dispositivos mobiles já possuem aplicativos para leitura, causam desinteresse principalmente nos nativos digitais que estão acostumados com conteúdo multimídia. Esta pesquisa apresenta uma solução de uma plataforma mobile para montagem de autoria, com intuito de implementá-la e efetivá-la totalmente através do celular ou tablet, o que atende as aspirações intrínsecas desta geração, seguindo os fundamentos de se evitar a sobrecarga cognitiva.

Palavras – chave: AVA, EAD, plataforma de ensino, mobile, autoria, sobrecarga cognitiva, m-Learning.

ABSTRACT

The increasing use of mobile technology, before by the generation Y today by digital natives, also by the public of the third age, has opened a gap, a gap between teachers and students about the available educational materials in hard copy and even in the cloud. E-books, especially for teaching platforms / learning in VLEs, EAD in .epub format, .mobi and .pdf whose mobile devices already have applications for reading, cause disinterest especially in digital natives who are accustomed to multimedia content. This research presents a solution of a mobile platform assembly authoring, with the intention of implementing it and making it fully through the mobile phone or tablet, that meets the intrinsic aspirations of this generation, following the fundamentals of avoiding cognitive overhead.

Key words: VLE, EAD, teaching platform, mobile, authoring, cognitive overload, m-Learning.

LISTA DE ILUSTRAÇÕES

| | |
|--|----|
| FIGURA 1 - The FRAME Model | 22 |
| FIGURA 2 – Código em Corona | 25 |
| FIGURA 3 – Código em Objective-C | 26 |
| FIGURA 4 – Código em Java | 27 |
| FIGURA 5 - Codificação Colocar Imagem | 37 |
| FIGURA 6 - Codificação Colocar | 38 |
| FIGURA 7 - Codificação Colocar Botão | 39 |
| FIGURA 8 - Codificação Colocar Vídeo | 40 |
| FIGURA 9 - Codificação Colocar Animação | 41 |
| FIGURA 10 - Codificação Colocar Plano de Fundo | 42 |
| FIGURA 11 - Codificação Colocar Texto | 44 |
| FIGURA 12 - Codificação Colocar Questão | 46 |
| FIGURA 13 - Codificação Criar Curso | 48 |
| FIGURA 14 - Codificação Criar Paginação | 49 |
| FIGURA 15 - Codificação Funções Auxiliares | 50 |
| FIGURA 16 - Codificação Cabeçalho Biblioteca | 50 |
| FIGURA 17 - Codificação Chamada Biblioteca | 50 |
| FIGURA 18 - Ferramenta Inicial Montador | 52 |
| FIGURA 19 – MCE Estrutura do Montador de Curso | 52 |
| FIGURA 20 – Ferramenta Colocar Texto | 53 |

| | |
|---|----|
| FIGURA 21 - Ferramenta Colocar Questão | 54 |
| FIGURA 22 – MCE Colocar um Texto | 55 |
| FIGURA 23 - Ferramenta Colocar Imagem | 55 |
| FIGURA 24 – MCE Colocar uma Imagem | 56 |
| FIGURA 25 - Ferramenta Colocar Som | 57 |
| FIGURA 26 – MCE Colocar um Som | 57 |
| FIGURA 27 - Ferramenta Colocar Vídeo | 58 |
| FIGURA 28 – MCE Colocar um Vídeo | 58 |
| FIGURA 29 - Ferramenta Colocar Animação | 59 |
| FIGURA 30 – MCE Colocar Animação | 60 |
| FIGURA 31 – Dispositivos Mobiles Corona | 60 |
| FIGURA 32 – Build Aplicação Corona SDK | 61 |

SUMÁRIO

| | |
|---|-----------|
| 1 INTRODUÇÃO | 11 |
| 1.1 Contextualização | 11 |
| 1.1.1 Tecnologia e ensino a distância | 12 |
| 1.1.2 Processo cognitivo com aprendizagem em ambiente virtual | 12 |
| 1.1.3 Nativos digitais e os seus dispositivos móveis | 14 |
| 1.2 Caracterização geral do estudo | 14 |
| 1.2.1 Questões de investigação e objetivos do estudo | 15 |
| 1.2.2 A importância de um montador de autoria | 18 |
| 1.2.3 Apresentação do m-Learning | 18 |
| 1.2.4 A adaptação do m-Learning junto ao processo cognitivo | 19 |
| 1.3 Terminologia usada | 23 |
| 2 CORONA SDK | 25 |
| 3 PLATAFORMAS | 29 |
| 3.1 E-books – Kindle | 29 |
| 3.2 Tablets – iOS e Android | 30 |
| 3.3 iOS | 31 |
| 3.4 Android | 32 |
| 3.5 Windows Phone | 32 |
| 3.6 Não existe sistema ideal | 33 |
| 4 METODOLOGIA E PESQUISA | 34 |
| 4.1 Biblioteca Educação a Distância (EAD) | 35 |
| 4.1.1 Função Colocar Imagem | 36 |
| 4.1.1.1 Codificação em CoronaSdk-Lua (biblioteca libEA.lua) | 37 |
| 4.1.2 Função Colocar Som | 37 |
| 4.1.2.1 Codificação em CoronaSdk-Lua (biblioteca libEA.lua) | 38 |
| 4.1.3 Função Colocar Botão | 38 |
| 4.1.3.1 Codificação em CoronaSdk-Lua (biblioteca libEA.lua) | 39 |
| 4.1.4 Função Colocar Vídeo | 40 |

| | |
|---|-----------|
| 4.1.4.1 Codificação em CoronaSdk-Lua (biblioteca libEA.lua)..... | 40 |
| 4.1.5 Função Colocar Animação | 40 |
| 4.1.5.1 Codificação em CoronaSdk-Lua (biblioteca libEA.lua)..... | 41 |
| 4.1.6 Função Colocar Plano de Fundo..... | 42 |
| 4.1.6.1 Codificação em CoronaSdk-Lua (biblioteca libEA.lua)..... | 42 |
| 4.1.7 Função Colocar Texto | 43 |
| 4.1.7.1 Codificação em CoronaSdk-Lua (biblioteca libEA.lua)..... | 44 |
| 4.1.8 Função Colocar Questão..... | 44 |
| 4.1.8.1 Codificação em CoronaSdk-Lua (biblioteca libEA.lua)..... | 46 |
| 4.1.9 Criar Curso | 47 |
| 4.1.9.1 Codificação em CoronaSdk-Lua (biblioteca libEA.lua)..... | 48 |
| 4.1.10 Função Criar Paginação..... | 48 |
| 4.1.10.1 Codificação em CoronaSdk-Lua (biblioteca libEA.lua)..... | 49 |
| 4.1.11 Funções Auxiliares, Cabeçalho da Biblioteca e Chamada da Biblioteca no Programa Principal | 50 |
| 4.1.11.1 Funções Auxiliares | 50 |
| 4.1.11.2 Cabeçalho da Biblioteca | 50 |
| 4.1.11.3 Chamada da Biblioteca no Programa Principal a ser criado (main.lua) | 50 |
| 5 MONTADORES | 51 |
| 5.1 Etapa inicial do montador de autoria..... | 51 |
| 5.2 Apresentação de textos | 53 |
| 5.3 Inserir imagem | 55 |
| 5.4 Colocar som | 56 |
| 5.5 Inserir vídeo | 58 |
| 5.6 Gerar Animação..... | 59 |
| 5.7 Finalizar e pré-visualizar o trabalho | 60 |
| 6 CONCLUSÃO E RELEVÂNCIA | 62 |
| 6.1 Trabalhos futuros | 63 |
| REFERÊNCIAS..... | 64 |

1 INTRODUÇÃO

Este capítulo inicia-se com uma introdução, em que se procede à contextualização do tema (1.1), aborda o uso da tecnologia para Ensino a Distância (EAD) (1.1.1), apresentam-se os princípios do processo cognitivo de aprendizagem em ambiente virtual (1.1.2) e caracterizam-se os nativos digitais e os seus dispositivos móveis (1.1.3). Segue-se a caracterização geral do estudo (1.2), a apresentação das questões de investigação e os objetivos da pesquisa (1.2.1), a importância de um montador de autoria a ser proposto (1.2.2), a apresentação do m-Learning (1.2.3) e a sua adaptação junto ao processo cognitivo (1.2.4). Enfim, encerra-se o capítulo expondo a terminologia usada (1.3).

1.1 Contextualização

Com o desenvolvimento e proliferação de novas tecnologias, muita investigação tem sido realizada, nos últimos tempos, sobre como as pessoas usam as inovações tecnológicas. Em virtude da evolução das tecnologias móveis, surgiu uma preocupação em explorar a forma como estas tecnologias podem ser mais bem utilizadas como ferramentas de aprendizagem (KUKULSKA-HULME, 2009) e como o seu desenvolvimento e utilização têm sido moldados por fatores sociais e culturais (WAYCOTT; HULME, 2003).

O foco deste estudo é adaptar essa inovação para o Ensino a Distância (EAD) e proporcionar o melhor entendimento ao aprendiz, através de um conteúdo que possa ser passado de forma simples e inovadora pelo educador. Para obtenção de resultados a elaboração de um motor para cursos de EAD, realizou-se através de uma ferramenta de desenvolvimento de software (SDK – Software Development Kit) conhecida como Corona SDK. Como será abordado mais a frente, trata-se de um recurso em que se oferece um ambiente de desenvolvimento para programar aplicativos para dispositivos mobiles. Através desta plataforma foi proposta uma biblioteca, criada para este trabalho, que irá compor todas as funções básicas, servindo como “engrenagens” deste motor para apresentação de um modelo de ensino que atenda o propósito desta dissertação.

Para tanto, será abordado neste trabalho, desde as inovações tecnológicas com dispositivos móveis, com ferramentas que nos auxiliam no ensino, até a maneira de como podem ser utilizadas pela forma cognitivamente mais eficaz.

1.1.1 Tecnologia e ensino a distância

O aprendizado com quadro negro, livros e uma sala de aula até hoje é eficiente e proliferado em todo mundo, mas constantemente procura-se inovar a sua abordagem de ensino frente ao aluno, para que este possa interagir com o que está sendo ensinado e assimilar o conhecimento apresentado. Para isto, com forte frente à educação, tem-se no mundo todo tecnologias plausíveis em sistemas virtuais que atendem a esses requisitos de inovação ao ensino.

A EAD consolidou-se com sucessivas inovações, na aquisição de softwares e plataformas de gerenciamento que sustentam o crescimento e o desenvolvimento de cursos cada vez mais elaborados dentro da própria organização. O que promove este crescimento são os Sistemas Virtuais de Aprendizagens (AVA), identificados como softwares que auxiliam na montagem de cursos acessíveis, sejam eles, comumente na internet, elaborados para ajudar educadores na gestão de conteúdos aos aprendizes e, presencialmente, para complementação em salas de aulas.

Importante levantar a questão de como um ambiente de aprendizagem virtual pode agregar as pessoas, por meio da EAD, pois, dentre muitos, alguns objetivam somente o entretenimento, seguidos de outros voltados à divulgação de notícias e até que, por fim, surgem aqueles focados no sistema de ensino e aprendizagem pela internet. Para potencializar um sistema de aprendizado virtual é necessário explorar estes ambientes, que são definidos da seguinte forma:

Ambientes digitais de aprendizagem são sistemas computacionais disponíveis na internet, destinados ao suporte de atividades mediadas pelas tecnologias de informação e comunicação. Permitem integrar múltiplas mídias, linguagens e recursos, apresentar informações de maneira organizada, desenvolver interações entre pessoas e objetos de conhecimento, elaborar e socializar produções tendo em vista atingir determinados objetivos (ALMEIDA, 2003, p. 331).

No entanto, não há restrição quanto ao uso destas ferramentas por educadores e nem um caminho da melhor forma de utilizá-las. Assim, é possível considerar o processo cognitivo como ponto central para trabalhar em um sistema adequado a estes ambientes.

1.1.2 Processo cognitivo com aprendizagem em ambiente virtual

Uma das grandes vantagens de um Sistema Virtual de Aprendizagem (AVA, m-

Learning ou EAD) é que se pode controlar especificamente o que se deseja que o aprendiz veja a cada momento. O conceito básico é implementar um sistema que tutele, direcione o profissional do ensino a montar sistemas de apresentação de conteúdo multimídia que favoreça o processo de representação mental adequada (MAYER, 2001) de forma que o aprendiz tenha um foco definido a cada apresentação, evitando a sobrecarga multimídia (PAAS; RENKL; SWELLER, 2003). A virtualização do ambiente de aprendizagem vem agregar à aquisição de conhecimento com maior dinamismo, rapidez e com foco nos objetivos entre seus participantes (LEVY, 1996).

Entretanto, isto precisa estar alinhado com o processo cognitivo do ser humano, o qual se refere ao estudo do processamento humano de informações, ou seja, o estudo de como os seres humanos percebem, processam, codificam, estocam, recuperam e utilizam as informações. A estrutura cognitiva humana inclui três sistemas de memória: a memória sensorial, a memória de curta duração e a memória de longa duração, as quais trabalham em conjunto (NUNES; GIRAFFA, 2003). Dessa maneira, é importante que um sistema AVA caminhe junto com este processo.

Segundo Santos e Tarouco (2007), os materiais educacionais que fazem uso de recursos digitais, frequentemente, sobrecarregam a memória de trabalho dos aprendizes e dificultam a aquisição de esquemas que requerem reflexão. Uma análise baseada nos princípios da Teoria da Carga Cognitiva, em que Paas, Renkl e Sweller (2003) definem como um conjunto universal de princípios que resultam em um ambiente de aprendizagem eficiente e que, consequentemente, promovem um aumento na capacidade do processo de cognição humana, só reforça a ideia de que os recursos tecnológicos, por mais atraentes que possam parecer, nem sempre estão de acordo com o processo cognitivo humano e, por isto, nem sempre qualificam o processo de aprendizagem.

Essa teoria, de acordo com Santos e Tarouco (2007), aplica-se a todos os tipos de conteúdos, todos os tipos de mídias e à todos os estudantes, visto que ela tem como fim elaborar as ferramentas de ensino – texto, imagens e áudio – e aplicá-las a todo o conteúdo de ensino, bem como, às plataformas de aprendizagem a distância, no intuito de potencializar a aprendizagem e desenvolver habilidades flexíveis, através da criação e uso de recursos e ambientes de aprendizagem, que estejam em sintonia com o processo cognitivo humano.

Segundo Paas, Renkl e Sweller (2003), a aprendizagem acontece de maneira melhor quando o processo de informação está alinhado com o processo cognitivo humano, ou seja, quando o volume de informações oferecidas ao aprendiz for compatível com a capacidade de compreensão humana. Assim, é necessário considerar, conforme Miller (1956) aponta a

impossibilidade natural do ser humano em processar muitas informações na memória a cada momento.

1.1.3 Nativos digitais e os seus dispositivos móveis

É importante esclarecer, mesmo brevemente, como os aparelhos digitais estão sendo usados pelos estudantes para facilitar ou obstruir seu processo de aquisição de conhecimento e entender como eles estão interagindo com colegas e educadores quando a relação é mediada por tais aparelhos.

Identificar como a nova geração de aprendizes (Geração Y) aprende e utiliza as tecnologias digitais de informação e comunicação e analisar como o domínio desses processos afeta a aprendizagem escolar desta geração, são ações indispensáveis que poderão auxiliar o docente a reprogramar conteúdos e a planejar sua prática pedagógica, a fim de satisfazer às necessidades e expectativas dos aprendizes contemporâneos. (XAVIER, 2011).

O termo nativos digitais, atualmente, condiz mais com a caracterização desta geração. Araújo Júnior et al. (2016) explica o motivo de não utilizarmos o termo Geração Y, pois esta não nasceu com a tecnologia no berço, mas a adquiriu desde jovem, já os Nativos Digitais nascem com um tablet como babá em um espaço já pronto para uso no próprio berço. As soluções mobiles fazem parte do corpo do nativo digital, como um braço, ou uma perna ou como algo imprescindível na vida destes.

Dessa forma, essa geração “digitalizada” está voltada para o uso de microcomputadores, utilizando os aparelhos digitais com muita desenvoltura, em especial, o computador e o celular, cotidianamente, durante várias horas, tanto que, para muitos, eles se tornaram um apêndice em seu dia a dia. Desvelar a influência desses aparelhos de uso frequente na vida e na aprendizagem dessa geração configura uma motivação suficiente e necessária para a realização de um empreendimento acadêmico.

1.2 Caracterização geral do estudo

Consideradas as pequenas frações de tempo, ao final de um dia, pode-se perceber que grande parte dele foi dedicado ao uso do próprio smartphone. Brasileiros estão no ranking dos que mais acessam dispositivos móveis em todo o mundo. Segundo uma pesquisa feita em 2013, pelo IBOPE, em parceria com o Instituto Conecta e a Worldwide Independent Network

of Market Research (WIN), o brasileiro passa, em média, cerca de 84 minutos por dia usando o smartphone e 79 minutos usando tablets (HECKE, 2013). Este número supera a média global que é de 10 minutos e vem aumentando anualmente.

A tecnologia mobile cresce mais rápido do que a internet cresceu na década de 90. Algumas estatísticas apontam que já existem mais smartphones do que televisores no mundo e, em alguns países, o número de aparelhos já é maior do que a sua população. No Brasil, os smartphones estão entre os produtos de consumo mais vendidos.

Atualmente, 15% do tráfego global na internet são provenientes de dispositivos móveis e cerca de 50% das buscas acontecem via mobile. Por ano, são investidos cerca de 22 bilhões de dólares em propagandas para o público mobile, e este número só aumenta, principalmente, porque mais de 15% das compras de varejo já são realizadas através destes dispositivos (OLIVEIRA, 2016).

Diante disso, é conveniente aproveitar os dispositivos mobiles, celulares e tablets, cada vez mais disponíveis, para possibilitar novas formas de aprendizado. Assim, é importante o desenvolvimento de novas formas que permitam o estudo tão vantajoso quanto estudar a partir de um computador. Muitas universidades, cursos e empresas estão apostando no ensino através de dispositivos móveis, como atualmente é conhecido, o mobile learning (m-Learning).

Não se pode prescindir, atualmente, da multimídia, já que os dispositivos mobiles abusam desta tecnologia, e, se não utilizadas, causam desinteresse nos alunos nativos digitais e até mesmo na geração Y, que são a maioria dos alunos de quatro a trinta e cinco anos (FERREIRA et al., 2014).

Em uma sociedade competitiva, na qual se interage diariamente com os mais variados recursos tecnológicos, seja em casa, no trabalho ou no lazer, em que a valorização do conhecimento está cada vez mais significativa, pode-se fazer uso dos princípios da Teoria da Carga Cognitiva Máxima (que não cause sobrecarga), proposta por Paas, Renkl e Sweller (2003), para potencializar o processo de aprendizagem e de interação com a tecnologia.

A disponibilização de vários dispositivos móveis e o fácil acesso que atualmente são oferecidos ajudam a impulsionar a utilização destes para auxiliar na disseminação da educação. Assim, o bom aproveitamento da tecnologia é fundamental para a EAD.

1.2.1 Questões de investigação e objetivos do estudo

Com o crescente número da utilização de smartphones, já se tem mais de um por habitante do planeta, e o Brasil, com aproximadamente 210 milhões de habitantes, não é exceção, com aproximadamente 256 milhões de celulares, de acordo com a TELECO (Inteligência em Telecomunicações), tornando-se uma mídia desejável e oportuna para a EAD.

Os celulares e tablets já possuem aplicativos para leitura de e-books em formatos de arquivos mais comuns como **.epub**, **.mobi** e **.pdf**, existindo um vasto material educacional nesses tipos. Entretanto, mesmo com a portabilidade e a facilidade de ter em mãos o material quando desejar existe alguns fatores que não atraem tanto o público almejado, pois, infelizmente, os formatos citados são limitados quanto aos recursos multimídia e fracos na interatividade com o leitor.

Para se dotar os e-books, um detalhe que faz toda a diferença refere-se ao conteúdo multimídia (DIAS; COSTA; LIMA, 2009), pois um material com maior interação exige o desenvolvimento de um sistema que gere um novo formato de e-book, acessível a todas as plataformas mobile e desktop, contendo os recursos desejados e imprescindíveis para atrair a atenção dos leitores (alunos) e para repassar a informação de uma forma mais completa possível (áudio, vídeo, imagens, links, animações, hipertextos).

Pode-se pensar que já existem tais recursos para serem utilizados, principalmente, em plataformas desktop e nuvem, o que é verdade, mas, a maior lacuna, está no preparo de novos profissionais de autoria para programação mobile multi-device e multi-plataforma. O recurso de usar a nuvem nem sempre é viável, uma triste realidade brasileira, uma vez que, com exceção de algumas capitais, a internet é lenta e a banda larga acima de dois Megabytes/s não está ao alcance para a maioria da população. Para tanto, um sistema multimídia para conteúdo necessitaria para trabalhar em nuvem, satisfatoriamente, que se tivessem links com ou acima de 10 Megabytes/s.

A criação (projeto e implementação) de uma solução de um montador de autoria multimídia mobile, com motor de autoria instalado, via applet, no próprio dispositivo, viabilizaria o processo e evitaria a dependência da largura de banda e acesso à internet tanto para a montagem dos cursos quanto para quem fosse fazê-los.

Objetiva-se, em uma visão mais macro, permitir que autores, geradores de autoria, possam desenvolver seus cursos direto dos celulares e tablets, produzindo conteúdo que possa ser acessado em tempo real pelos internautas, como tem sido almejado nas m-Learnings.

Além disso, propõe-se um sistema no qual o motor processará somente a informação necessária para potencializar o aprendizado da informação fornecida. Os autores, dessa forma,

gerarão somente o que for relevante para a carga cognitiva eficaz. Assim, o princípio básico que motiva e justifica o sistema a ser proposto, como sendo efetivo no apoio ao ensino, apoia-se no Princípio da Navalha de Ockham (*Lex Parsinomiae*) (OCKHAM, 1999), que conclui pelas entidades e ações não devem ser multiplicadas além da necessidade. O princípio revela que a explicação para qualquer fenômeno deve assumir apenas as premissas estritamente necessárias à explicação do mesmo e eliminar todas as que não causariam qualquer diferença aparente nas predições da hipótese ou teoria.

Essa *lex parsimoniae* diz que o número de regras de inferência (no caso, ações metodológicas e ferramentas de ensino) somado ao número de axiomas (premissas) deve ser mínimo, portanto, para o nosso objetivo, de propor um montador que permite oferecer um material que explore da melhor forma o processo cognitivo, restringir apenas as informações necessárias a cada etapa é fundamental para evitar a sobrecarga multimídia.

O montador visa obedecer a Lei de Miller, na qual uma pessoa não consegue, ao mesmo tempo, memorizar e prestar atenção a mais de 7+/- 2 chuncks de informação na memória de curto prazo (MILLER, 1956), e a restrição imposta por Cowan (2000), em que reduz os Chunks de Miller (7+/-2) a quatro para adultos e jovens, e menos para crianças.

Ao seguir esta linha para a implementação de um material digital, o aprendiz terá mais facilidade na assimilação da informação apresentada, adquirindo o conhecimento. Estes estudos são comprovados, também, pelas experiências em degradação cognitiva, realizadas pelo neurocientista Raz, Fan e Posner (2005) e pelas experiências de Strayer et al. (2013), em laboratórios da Universidade de Utah (USA), em que afirmam que não se pode esquecer que apenas um processo cognitivo pode ser disparado no aluno a cada etapa do curso, já que o cérebro não foi feito para ser multitarefas (MEDINA, 2009).

Isso elimina a arbitrariedade e quebra da transparência referencial no processo cognitivo, analogamente, um conceito comum na matemática, em que dada uma função ou método podemos substituí-la pelo seu valor de retorno sem causar impacto na aplicação, ou seja, um processo cognitivo disparado em uma etapa é como um resultado determinado unicamente pelos seus valores de entrada, ou seja, nada no mundo externo, além da entrada, pode afetar sua saída. A informação não sofrerá desvio no seu resultado para quem está recebendo-a.

Ao focar estes conceitos voltados ao ensino, significa que as informações contidas no material didático, tangível ou digital, a cada passo, nunca devem evocar no aluno mais de uma inferência, mais de um processo cognitivo, ou seja, sempre deve evocar o mesmo aprendizado em qualquer pessoa que o tenha de posse para estudo.

1.2.2 A importância de um montador de autoria

Um dos intuios desta pesquisa, como já foi apresentado, é gerar um montador de autoria mobile que tutele, de forma transparente ao conteudista, a montagem de um curso que efetive os objetivos de aprendizagem traçados inicialmente no curso, evitando resultados arbitrários e ambíguos, isto é, obtendo resultados matematicamente ortogonais (um ponto em um dos eixos não tem projeção nos demais, ou seja, possui apenas uma interpretação, portanto, não arbitrária).

Ao visualizar, de uma forma mais simples, um ponto no espaço não apresenta uma solução ortogonal apenas, mas projeções nos eixos x, y e z. Desta forma, um mesmo ponto pode apresentar visões distintas em cada eixo e, também, nos planos xy, xz e yz. Cada uma das projeções são parcelas da informação do que se quer transmitir, com visões diferentes, mas que, sozinhas, não traduzem a realidade como um todo. No que diz respeito ao ensino, soluções ortogonais devem ser aquelas que pontuam apenas nos eixos, assim, um ponto no eixo x não tem projeção no eixo y e no eixo z, bem como nos planos xy, xz e yz. Consequentemente, só se pode abstrair um e apenas um conhecimento do que se está ensinando, sem dualidades de interpretação e/ou resultados arbitrários.

O sistema projetado visa, de uma forma transparente ao conteudista, como já dito, conduzir o montador a produzir conteúdos ortogonais e sem sobrecarga multimídia cognitiva, simultaneamente, atrativos e interativos, conforme aderência aos nativos digitais usuários de soluções mobiles.

1.2.3 Apresentação do m-Learning

O Mobile Learning, também conhecido como m-Learning, é derivado do ensino a distância, e pode ser definido como o uso de tecnologias móveis como smartphones, tablets e iPods para obter uma experiência de aprendizado.

Esse modelo proporciona aos alunos a capacidade de aprender utilizando a mobilidade ao seu favor, com acesso imediato e ilimitado à informação, na hora e no local em que necessitam. Concomitantemente, é uma experiência caracterizada, que permite trabalhar em condições adequadas para o aprendizado eficaz.

Para destacar as vantagens e o motivo da utilização do modelo, apresentam-se alguns dos pontos fortes do m-Learning, de acordo com a empresa Sambatech (2013):

- **Portabilidade:** Tecnologia que permite ser executada de qualquer tipo de device, e pode ser usada em diferentes tipos de sistema.
- **Interatividade:** dá ao aluno a possibilidade de interagir, ou seja, de realizar uma ação integrada, e se envolver na busca do conhecimento e na experiência de aprendizado.
- **Sensibilidade ao contexto:** há uma grande quantidade de recursos que o mobile possui e oferece aos educadores a possibilidade de explorá-los.
- **Conectividade:** Capacidade de se conectar a internet, realizar ações ou fazer operações em um ambiente de rede.
- **Individualidade:** Cada aluno constrói a sua experiência individual, respeitando o seu tempo, espaço e também a sua forma de aprender.

Com esses pontos, elaborou-se para esta pesquisa uma prototipação adequada em repassar a melhor ideia do conhecimento a ser aprendido pelo estudante. Para tanto, foram consideradas a familiaridade e o acesso a dispositivos móveis destes estudantes, o tipo de conteúdo a ser passado, durável e denso para memorização (o que é uma característica do eLearning) e propício para que alunos o tenham sempre à disposição quando precisarem, outra vantagem do m-Learning.

Através desses pontos, um curso poderá ser criado por um App para um modelo de EAD e facilitar o aprendizado e engajamento dos educandos. O m-Learning não se limita apenas a pensar em cursos, treinamentos e aulas realizadas através de dispositivos mobile. Ele abrange o vasto ambiente do EAD, porém, adaptado aos dispositivos móveis. É importante entender que o conteúdo deve ser pensado, desde o início, em atender a realidade do mobile e para o modo como as pessoas irão consumi-lo.

1.2.4 A adaptação do m-Learning junto ao processo cognitivo

Os alunos, hoje, estão pouco interessados em passar, uma parte do seu tempo, sentados numa cadeira dentro de uma sala de aula, preferindo substituir as aulas por discussões formais ou informais com os seus colegas em pequenos grupos em ambientes virtuais (JOHNSON, 2008; JOHNSON; LOMAS, 2005).

Quando se investe em tecnologia para a sala de aula, é necessário mudar a arquitetura

do ensino baseado na transmissão ou exposição, em que o professor transmite informação para uma audiência de alunos. Os ambientes com uma estrutura tradicional, por vezes, dificultam a interatividade, a colaboração, o trabalho de grupo e a construção colaborativa que as tecnologias propiciam. Não é possível adaptar um modelo tecnológico e continuar a funcionar de forma tradicional, porque na opinião de Johnson (2008) e Johnson e Lomas (2005, p. 22), “New technologies and their adoption have always had an influence on what happens in the classroom”. Para os autores ainda, as tecnologias com wireless permitem a alunos e professores encontrar novas formas de comunicar, colaborar e interagir. Por isso, é necessário apostar mais em espaços de aprendizagem do que em salas de aulas.

As atividades educativas suportadas por tecnologia permitem aos alunos maior autonomia e participação e exigem novos espaços de aprendizagem. Para isso, Johnson (2008) e Johnson e Lomas (2005) consideraram os seguintes princípios:

- As atividades de aprendizagem ocorrem em maior proporção fora da sala de aula;
- A sala de aula deve ser flexível, com mobiliário móvel, para permitir configurações adequadas às diferentes atividades e disciplinas;
- Os espaços de aprendizagem devem ser pensados como um ecossistema interconectado;
- A tecnologia deve ser padronizada para facilitar a operação;
- Devem-se criar espaços de aprendizagem tanto informais, como virtuais.

Na educação para o século XXI a aposta deve ser num currículo atual, já que os papéis de alunos e professores podem-se inverter, porque, em determinado momento, o aluno também pode ser o expositor (conteúdo criado) ou planificador da sua própria aprendizagem. Conforme corroboram Johnson (2008) e Johnson e Lomas (2005), muitos dos alunos estão mais confortáveis com a tecnologia do que os seus professores e alguns alunos parecem preferir comunicar usando mensagens instantâneas a estar nas aulas. Frente aos novos modos de interação, comunicação e socialização, novos modelos educacionais devem ser pensados e, consequentemente, impõe-se em repensar espaços como a escola, universidades ou qualquer centro de ensino.

Com uma rápida observação, podem-se perceber como muitos indivíduos estão sempre com os celulares à mão. Dessa maneira, este trabalho levanta a questão de como tirar vantagem desta tendência e oferecer novas formas de aprendizado, utilizando a estratégia de

mobile learning.

Muitas universidades e instituições de ensino estão em busca de um método capaz de engajar alunos e o mobile learning, modelo de ensino através de dispositivos móveis, oferece a capacidade de aprender utilizando a mobilidade ao seu favor, com acesso imediato e ilimitado à informação, na hora e no local em que necessitam.

Para explorar de forma mais detalhada o mobile learning, levantaram-se as seguintes diretrizes para elaborar um projeto que busque atingir o objetivo de propagar a informação neste processo:

- Implantar o mobile learning como um curso
- Otimizar resultados associando o microlearning com o mobile learning
- Aproveitar o máximo do processo cognitivo do aluno
- Melhorar o engajamento com os alunos
- Qualquer educador implementar em sua realidade com uso de mobile

Com estes pontos busca-se alterar o tempo de estudo do aluno, pois a retenção deste será menor, uma vez que o curso será trabalhado no formato microlearning, ou seja, pequenas unidades de conteúdo possibilitarão que o aluno possa estudar, em curtos intervalos de tempo, com maior foco. Através deste método, novos espaços de aprendizagem emergem, tornando-o mais longo durante a vida (INNSBRUCK, 2005).

O modelo de Koole (2009), Ph.D. in E-Research and Technology-Enhanced Learning 2013, Centre for Technology-Enhanced Learning, Lancaster University, UK, aborda questões pedagógicas contemporâneas de sobrecarga de informações, conhecimentos de navegação e colaboração na aprendizagem. Este modelo é útil para guiar o desenvolvimento de futuros dispositivos móveis, de materiais de aprendizagem, e o desenho de estratégias de ensino e de educação móvel de aprendizagem. Sua proposta está voltada para um Framework de Análise Reacional de Educação Mobile (FRAME).

O modelo FRAME leva em consideração as características técnicas dos dispositivos mobile, bem como os aspectos sociais e pessoais de aprendizagem (KOOLE, 2009). Neste modelo, experiências com mobile learning são vistas como existentes dentro de um contexto de informação em que, coletivamente e individualmente, os alunos consomem e criam informação. Dentro deste contexto de informação, o modelo FRAME é representado por um diagrama de Venn em que três aspectos se interceptam.

A Model for Framing Mobile Learning

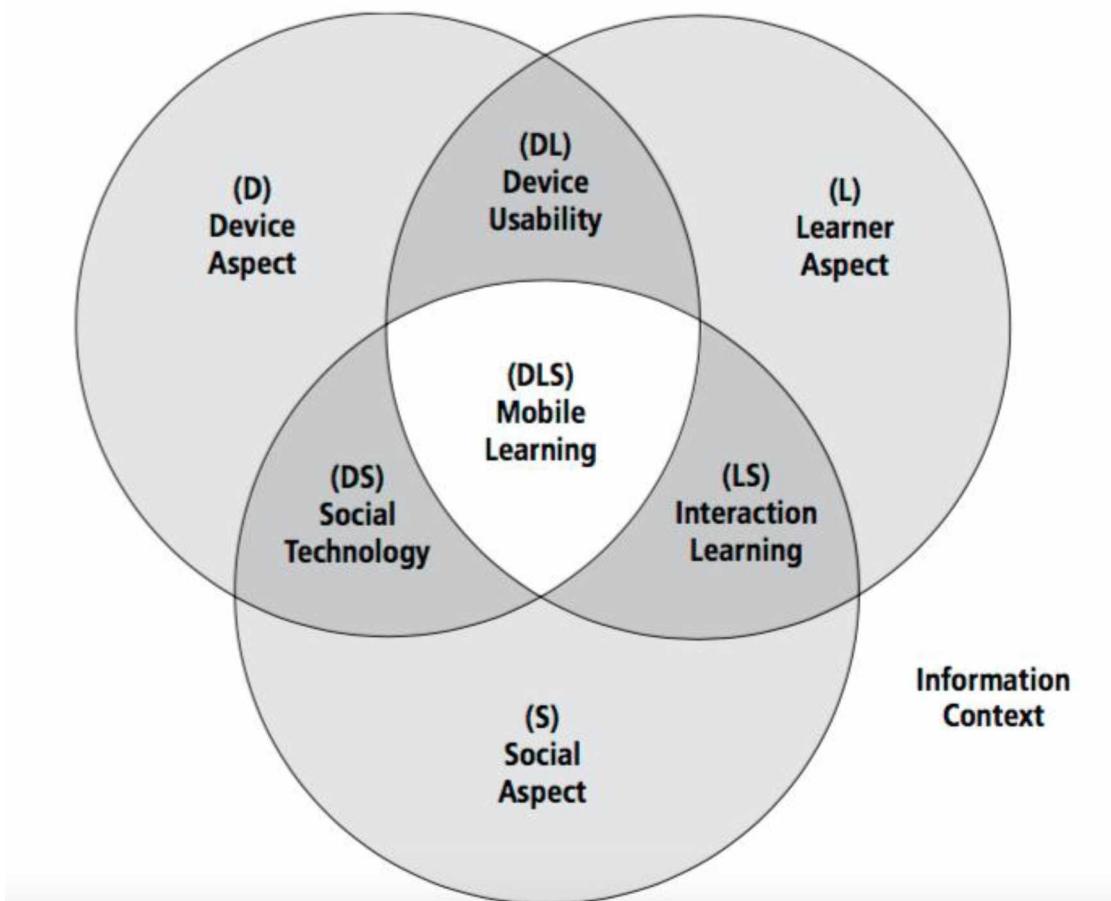


FIGURA 1 - The FRAME Model (Fonte: KOOLE, 2009).

Foram considerados aqui os três aspectos bases e o mobile learning process (DLS):

- **Aspecto do Dispositivo (D)**

O aspecto do dispositivo (D, Device aspect) referem-se aos aspectos físico, técnico e características funcionais de um dispositivo mobile.

- **Aspecto do Aprendiz (L)**

O aspecto do Aprendiz (L, Learner aspect) leva em consideração uma habilidade cognitiva individual como, memória, conhecimento prévio, emoções e possíveis motivações.

- **Aspecto Social (S)**

O aspecto social (S, Social aspect) leva em consideração o processo de interação social e cooperação.

- **Mobile Learning Process (DLS)**

Koole (2009) afirma que uma aprendizagem móvel eficaz consiste na interseção principal do modelo FRAME, resultante da integração do dispositivo (D, Device), aprendiz (L, Learner) e aspectos sociais (S, Social). Assim, é possível inferir que a aprendizagem com o uso de dispositivos móveis proporciona uma maior colaboração entre os alunos, acesso à informação e uma contextualização mais profunda de aprendizado.

Hipoteticamente, a aprendizagem móvel efetiva pode capacitar os alunos, permitindo-lhes avaliar e selecionar melhor uma informação relevante, redefinir seus objetivos, e reconsiderar a sua compreensão de conceitos dentro de um inconstante e crescente quadro de referência (o contexto da informação). Contudo, é importante fornecer um ambiente cognitivo melhorado em que alunos a distância possam interagir uns com os outros, com seus instrutores, com os materiais do curso e com seus ambientes, sejam físicos ou virtuais.

1.3 Terminologia usada

Como este estudo se insere num campo de pesquisa ainda recente, não há tradução para a língua portuguesa de grande parte dos termos utilizados em inglês, por isso, optou-se por mantê-los nas suas expressões originais.

Ao longo desta dissertação, os termos ferramentas, tecnologias e dispositivos são usados diferentemente para designar um aparelho eletrônico móvel como celulares (smartphones), tablets, e-books e outros dispositivos que as pessoas usam na sua vida pessoal e/ou para apoiar atividades de aprendizagem. Embora neste trabalho preferiu-se utilizar o termo dispositivo móvel para designar o celular, cabe frisar que não é o celular em si que se pretendeu avaliar nesta investigação, mas sim, os celulares em uso, enquanto ferramentas de aprendizagem.

Ademais, adotou-se nesta dissertação a expressão mobile learning ou m-Learning, em virtude de ainda não haver uma expressão em português que melhor traduza a expressão inglesa. Alguns estudos realizados no Brasil (MEIRELLES et al., 2006; REINHARD et al., 2005 apud ANDRADE et al., 2013) usam a expressão “aprendizagem com mobilidade” para designar o termo mobile learning, todavia, considerou-se que há uma dificuldade de adesão ao termo. Assim, ao discorrer sobre o assunto, as duas expressões foram empregadas, contudo, privilegiou-se o uso da expressão curta m-Learning, por analogia com o termo eLearning (e-

learning), comumente visto na literatura no âmbito da tecnologia educativa e ensino a distância.

Por último, vale destacar que a expressão “dispositivos móveis” refere-se aos aparelhos móveis que podem ser carregados na mão ou guardados no bolso e utilizados em qualquer lugar ou contexto. Uma série de dispositivos está incluída nesta terminologia, como os celulares, Tablets, Pockets PC (Laptops, Notebooks, Netbooks), e-books (Kindles), PSP, entre outros.

2 CORONA SDK

Corona é uma ferramenta de cross-compile e multi-plataforma usada para rapidamente criar aplicativos (apps) ou jogos. O que significa que pode ser criado um projeto e este poderá ser publicado e distribuído para diversos tipos de dispositivos, incluindo Apple (iPhone e iPad), Android (telefones e tablets), Amazon Fire, Mac desktop, Windows Desktop, e dispositivos para TVs tais como Apple TV, Fire TV, e Android TV.

Diferentemente da maioria das ferramentas de desenvolvimento de softwares que exigem uma programação distinta para cada projeto, o Corona oferece uma maneira adequada para ensinar o desenvolvimento de aplicativos, pois com um único projeto e uma única programação, diferentes plataformas, como IOS, Android e Windows Phone, podem ser utilizadas. Entre tantas opções para o desenvolvimento de aplicações multi-plataforma, como PhoneGap e Xamarin, o Corona claramente se destaca sobre os outros.

A decisão de usar Corona foi requerida pela capacidade da plataforma ser flexível e gerar aplicativos para tantas plataformas diferentes, enquanto outras SDKs concorrentes só permitem fazer apenas aplicativos ou jogos, não ambos. Usuários podem desenvolver aplicativos sofisticados e jogos em pouco tempo (<https://coronalabs.com/products/corona-is-10x-faster/>), como pode ser exemplificado nos trechos de código, nas figuras 2, 3 e 4.

```
1 --Display "myImage.png"  
2 display.newImage("myImage.png");
```

FIGURA 2 – Código em Corona (Fonte: Corona Labs).

```
1 // Display "myImage.png"
2 //
3 // -----
4 // OpenGLTextureAppDelegate.m
5 // -----
6
7 #import "OpenGLTextureAppDelegate.h"
8 #import "EAGLView.h"
9 #import "OpenGLTextureViewController.h"
10
11 @implementation OpenGLTextureAppDelegate
12
13 @synthesize window=_window;
14
15 @synthesize viewController=_viewController;
16
17 - (BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:(NSDictionary *)launchOptions
18 {
19     // Override point for customization after application launch.
20     self.window.rootViewController = self.viewController;
21     return YES;
22 }
23
24 - (void)applicationDidBecomeActive:(UIApplication *)application
25 {
26
27 ...
28
294     glClearColor(0.0f, 0.0f, 0.0f, 1.0f);
295     glClear(GL_COLOR_BUFFER_BIT);
296
297
298     glMatrixMode(GL_PROJECTION);
299     glLoadIdentity();
300     glMatrixMode(GL_MODELVIEW);
301     glLoadIdentity();
302
303     glVertexPointer(2, GL_FLOAT, 0, squareVertices);
304     glEnableClientState(GL_VERTEX_ARRAY);
305     glTexCoordPointer(2, GL_FLOAT, 0, texCoords);
306     glEnableClientState(GL_TEXTURE_COORD_ARRAY);
307
308     glDrawArrays(GL_TRIANGLE_STRIP, 0, 4);
309
310     [(EAGLView *)self.view presentFramebuffer];
311 }
312
313 @end';
```

FIGURA 3 – Código em Objective-C (Fonte: Corona Labs).

```

1  /** Display "myImage.png" */
2  package net.obviam.opengl;
3
4  import java.nio.ByteBuffer;
5  import java.nio.ByteOrder;
6  import java.nio.FloatBuffer;
7
8  import javax.microedition.khronos.opengles.GL10;
9
10 import android.content.Context;
11 import android.graphics.Bitmap;
12 import android.graphics.BitmapFactory;
13 import android.opengl.GLUtis;
14
15 public class Square {
16     private FloatBuffer vertexBuffer; // buffer holding the vertices
17     private float vertices[] = {
18         -1.0f, -1.0f, 0.0f, // V1 - bottom left
19         -1.0f, 1.0f, 0.0f, // V2 - top left
20         1.0f, -1.0f, 0.0f, // V3 - bottom right
21         1.0f, 1.0f, 0.0f // V4 - top right
22     };
23
24     private FloatBuffer textureBuffer; // buffer holding the texture coordinates
...
83     /** The draw method for the square with the GL context */
84     public void draw(GL10 gl) {
85         // bind the previously generated texture
86         gl.glBindTexture(GL10.GL_TEXTURE_2D, textures[0]);
87
88         // Point to our buffers
89         gl.glEnableClientState(GL10.GL_VERTEX_ARRAY);
90         gl.glEnableClientState(GL10.GL_TEXTURE_COORD_ARRAY);
91
92         // Set the face rotation
93         gl.glFrontFace(GL10.GL_CW);
94
95         // Point to our Vertex buffer
96         gl.glVertexPointer(3, GL10.GL_FLOAT, 0, vertexBuffer);
97         gl.glTexCoordPointer(2, GL10.GL_FLOAT, 0, textureBuffer);
98
99         // Draw the vertices as triangle strip
100        gl.glDrawArrays(GL10.GL_TRIANGLE_STRIP, 0, vertices.length / 3);
101
102        //Disable the client state before leaving
103        gl.glDisableClientState(GL10.GL_VERTEX_ARRAY);
104        gl.glDisableClientState(GL10.GL_TEXTURE_COORD_ARRAY);
105    }
106 }

```

FIGURA 4 – Código em Java (Fonte: Corona Labs).

A linguagem utilizada pelo Corona é a linguagem de script Lua que fornece uma grande opção de APIs e plugins, permitindo que os desenvolvedores rápido e facilmente criem aplicativos que possam ser executados em diversos tipos de dispositivos.

Na plataforma de desenvolvimento do Corona utiliza-se o Corona Simulator, o qual permite que se possa visualizar como uma app ficará em diferentes tipos de dispositivos e

como responderá em cada um, dispensando realizar um deploy do aplicativo em um dispositivo real para testá-la. Uma vez que o app estiver pronto para um teste em um dispositivo real, poderá ser gerado um build do aplicativo, compilando o código desenvolvido para a plataforma desejada e instalando no respectivo dispositivo, para no final ser testado antes de submetê-lo as marketplaces(*) de apps.

A ferramenta também conta com diversos plug-ins(*) os quais podem ser adicionados para específicas funcionalidades e ajudar na evolução do desenvolvimento de um aplicativo, com inovações de tecnologia para cada dispositivo, como GPS, Câmera e Vídeo, ou até mesmo, em facilitar o desenvolvimento para um determinado componente.

Para a monetização de um app, é possível incluir propagandas para aplicativos através do simples uso do Corona Ads ou um de seus diversos plug-ins. A ferramenta fornece ainda o Corona Cloud, em que serviços nas nuvens são utilizados para armazenar dados de jogos, de usuários e até mesmo de cursos online, como os que serão propostos neste trabalho. Além disso, a ferramenta sugerida fornece plug-ins para integração com redes Sociais como Facebook e Twitter.

Essencialmente, a plataforma Corona tem como objetivo auxiliar no desenvolvimento de uma aplicação com agilidade e eficiência, com o benefício de escrever uma codificação uma única vez e realizar o deploy(*) para múltiplas plataformas. Diante disso, a plataforma Corona foi uma escolha natural para o desenvolvimento deste trabalho.

O desenvolvimento na plataforma Corona com a linguagem Lua adere as necessidades desta pesquisa, uma vez que é uma linguagem escalável e de fácil aprendizado e possui as funcionalidades para elaboração de um montador de autoria. As bibliotecas do Corona, nesta mesma linguagem, permitem prototizarmos um montador com fácil acesso a inserção de mídias, sons, vídeos e imagens, além da formalização de textos. Com todo o recurso disponível para compilação em multi-plataformas, a utilização dos componentes de dispositivos móveis, é possível implementar um montador de um curso, no qual objetiva-se a instruir e facilitar, para o tutor, a elaboração de um material com a melhor forma de apresentar a informação ao conteudista.

Para essas apresentações, diversas são as plataformas mobile, em que é possível aplicar o montador e o próprio aplicativo do curso gerado por ele, que serão descritas no próximo capítulo.

3 PLATAFORMAS

Seguindo o conceito de m-Learning, o processo de ensino – aprendizagem deve ser idealizado de forma a permitir uma perfeita integração entre as ferramentas tecnológicas e as ações de aprendizagem utilizadas na educação. Existe uma composição harmoniosa entre o modelo conceitual de aprendizagem e as ferramentas tecnológicas utilizadas para atingir os objetivos da aprendizagem (RUGGIERO et al., 2005). Entre essas ferramentas, sobressaem os celulares, tablets e e-books, utilizados em larga escala pela população.

Cada um desses meios de dispositivos móveis possuem características peculiares e diferentes sistemas operacionais. Assim, procurou-se detalhar as características dos mais importantes e, portanto, dos mais utilizados, nos tópicos abaixo.

3.1 E-books – Kindle

Os livros eletrônicos ou e-books são uma realidade no mundo atual, ou seja, é possível encontrar na Web uma gama muito variada de exemplares disponíveis para consulta e/ou aquisição. Estes recursos potencializam uma maior difusão da informação e possibilitam a acessibilidade ao conhecimento nas partes mais remotas do planeta. De fato, os livros digitais podem ser acessados através de qualquer dispositivo eletrônico, seja o computador, o PDA (Patron Driven Acquisition), o telefone móvel, os smartphones, os iPhones e possuem alta portabilidade, já que podem ser armazenados num único dispositivo.

Segundo Mesquita e Conde (2008), o termo E-book tem origem inglesa e é uma abreviação para “electronic book”, ou livro eletrônico, pois se trata de uma obra com o mesmo conteúdo da versão impressa, com a exceção de ser uma mídia digital. O modelo eletrônico tem suas vantagens e portabilidade é uma de suas principais características, por exemplo, uma obra chinesa pode ser adquirida no Brasil, em questão de segundos. Se for considerada ainda a facilidade de transporte, enquanto milhares de e-Books podem ser facilmente manuseados com o uso de um dispositivo móvel (como um pendrive), carregar dois livros simultaneamente já pode ser trabalhoso.

Outra vantagem é que a facilidade de publicação e obtenção dos títulos digitais tem atraído vários autores e leitores originando a formação de uma nova classe editorial. O novo editor surge com alguns papéis que permanecem em relação a edição do livro impresso e outros inteiramente novos.

Em uma publicação, a Amazon.com anunciou que suas vendas de livros eletrônicos

superaram a venda de todos os livros de papel, sendo um marco na atual reconfiguração da cadeia produtiva dos livros, para escritores, editoras, distribuidores e leitores. Essa facilidade e o grande número de E-books utilizados pela população suscita uma discussão entre leitura impressa e digital.

A esse respeito, Lemos (2012) aborda que a leitura de um jornal ou de um livro em um tablet, como o iPad, por exemplo, não é nem como a leitura de um jornal impresso, nem como um jornal na web e nem como a leitura em um e-books (Kindle). Pois como uma análise da materialidade pode ser observada diversos agentes (dispositivos, produtores de textos, criadores de software, de imagens e de sons, usuários e suas práticas e hábitos corporais, lugares constituídos, distribuidores, escritores etc.) que atuam diferentemente a depender do conteúdo, do dispositivo e das práticas corporais. Ou seja, estes agentes proporcionam mais interação para se obter uma informação o que, neste estudo, vamos ressaltar que não pode ser benéfico ao aprendizado do leitor.

Para Medina e Freitas Filho (2004), a aprendizagem é significativa quando o estudante é capaz de relacionar os conhecimentos já adquiridos (ideias, conceitos e princípios) com a nova informação produzida, o que não acontece numa atividade de reprodução mecânica de conteúdos digitais disponíveis na Web. Os e-books nos proporcionam isso, mas sem as desvantagens de um livro ou jornal, como foi citado.

3.2 Tablets – iOS e Android

Os tablets são dispositivos portáteis que servem para acessar a internet, visualizar e editar textos e planilhas, jogar, tirar fotos, acessar músicas, vídeos, ler livros digitais (e-books) e outras coisas. Os tablets tem por características a tela sensível ao toque dos dedos “touch-screen” e, normalmente, suas dimensões variam entre 7 e 10 polegadas. Existem dois tipos de telas, as capacitivas e resistivas. O primeiro tipo apresenta muito mais sensibilidade ao toque e, portanto, uma melhor resposta, contudo, costumam ser mais caras, enquanto as telas resistivas, por serem compostas por várias camadas, possuem menor precisão no reconhecimento do toque, em contrapartida, apresentam custos reduzidos.

Os tablets utilizam aplicativos adaptados ao dispositivo. Lemos (2012) analisa que a informação é fixada em uma tela iluminada (bem diferente do conforto dos e-books), oferecendo a possibilidade de uma postura próxima daquela de quem lê um livro (e diferente daquela da web). Entretanto, o conteúdo pode ser outro, mais aberto e interativo, com links e

multimídias, adaptado à tela “touch-screen” e aos movimentos de rotação do equipamento, como os primeiros livros eletrônicos em CD-ROM. Por exemplo, ao usar o acelerômetro (rodá-lo e colocá-lo na posição horizontal ou vertical) uma imagem pode se transformar em um vídeo, por exemplo. A tela tátil permite uma interação mais complexa e intuitiva do que aquela com o teclado para a web, ou as teclas para passar as páginas de um e-book. A ação corporal é diferente daquela do impresso ou da web, e bem mais próxima dos e-books, embora a interatividade crie novas exigências de apoio do dispositivo ou de movimentos característicos.

A outra grande diferença entre os modelos disponíveis está no sistema operacional, que consistem, hoje, basicamente em três, a saber:

- iOS (sistema da Apple)
- Android (sistema do Google)
- Windows (sistema da Microsoft)

Nos tópicos a seguir, foi realizado, brevemente, um comparativo entre os sistemas operacionais iOS, Android e Windows Phone, para uma compreensão da melhor forma do desenvolvimento deste estudo (KIRCOVE, 2013) .

3.3 iOS

O iOS é conhecido como o SO que revolucionou o mundo dos celulares e trouxe uma nova visão para o mercado. Entre os seus modelos de dispositivos existem o iPhone, o iPod Touch e o iPad. Seus sistemas operacionais rodam com grande eficiência, devido, especialmente, a uma versão modificada do sistema operacional Mac OSX, que recebe o nome de iPhone OS. Seu intento é oferecer suporte para as tecnologias de reconhecimento de toques múltiplos, de inclinação (graças à inclusão do acelerômetro interno) e de multimídia e para a reprodução de vídeos, imagens e músicas, com o diferencial para as outras plataformas de possuir uma interface simplificada e uma usabilidade fácil e transparente.

Os aplicativos disponíveis só podem ser baixados na Apple Store. O lado positivo é que todos os downloads funcionarão, pois vírus ou outras ameaças com certeza não chegarão a este tipo de aparelho. Já um fator negativo é que sua interface é extremamente fechada, ou seja, o usuário quase não consegue fazer nenhuma personalização no sistema e encontra

dificuldades em utilizar acessórios de outros fabricantes no iPhone.

3.4 Android

O sistema operacional Android é o mais utilizado no mundo, já que sua plataforma é licenciável, o que significa que qualquer fabricante que atenda aos pré-requisitos do Google pode usá-lo em seus aparelhos. Por conta desse modelo, possui uma de suas maiores vantagens competitivas, este sistema atende a diversos tipos de usuário. Apesar da interface do Android ter muitas semelhanças com o iOS, o sistema do Google perde um pouco na facilidade de uso, fruto talvez da maior maturidade do iOS.

A loja virtual do Android, o Google Play, conta com milhares de aplicativos e se equipara à AppStore do iOS em volume. No entanto, devido ao seu modelo aberto, apresenta como ponto negativo o sistema ser mais suscetível a softwares maliciosos que podem danificar o aparelho. A variedade de aparelhos Android no mercado é um dos grandes trunfos do sistema, mas abriga outro um ponto negativo, pois nem todos os smartphones possuem a força necessária para atender as versões mais recentes do sistema operacional do Google. Assim, para garantir que o dispositivo atenda sem preocupações tudo que o sistema tem a oferecer, o melhor é comprar aparelhos mais aprimorados.

3.5 Windows Phone

O sistema operacional da Microsoft chegou atrasado à briga pelos smartphones, por isso sua participação é bem menor que a dos concorrentes no mercado. No entanto, isso não impediu que o sistema tivesse uma interface atrativa. Assim como o Android, o Windows Phone também é licenciável e está presente em diferentes modelos e marcas. Porém, há uma diferença fundamental, a Microsoft é mais exigente em termos de hardware de que o Google e, fazer isso, a companhia garante a fluidez do sistema operacional, mesmo em smartphones mais modestos, o que nem sempre acontece nos aparelhos Android.

Outro quesito que chama a atenção no Windows Phone é a sua interface. Ao contrário dos concorrentes Android e iOS sua aparência difere destes, podendo o usuário ter um pouco mais de dificuldade para se acostumar com sua usabilidade. Em termos de personalização, é possível apontar que é um intermediário entre os dois, nem tanto quanto o Android, nem tão pouco quanto o iOS. O ponto negativo do Windows Phone está em sua loja

de aplicativos, já que é um sistema mais recente e menos utilizado, a Windows Phone Store ainda é muito incipiente e não chega perto da oferta da App Store e do Google Play.

3.6 Não existe sistema ideal

Não é necessário definir o melhor sistema operacional para smartphones. Para este trabalho foram levantadas estas comparações, para demonstrar que a escolha do dispositivo a utilizar dependerá do perfil do usuário.

O iOS oferece estabilidade perfeita de integração entre hardware e software, design altamente intuitivo, mas por um alto custo de mercado. Já o Android é um sistema mais aberto com interface intuitiva, alto nível de personalização, variedade de modelos e preços. Por sua vez, o Windows Phone tem uma usabilidade mais complexa, contudo uma aparência familiar ao usuário, que se acostumou a utilizar a Microsoft.

Portanto, cabe ao usuário que irá preparar o curso analisar as especificidades dos sistemas que se enquadram ao curso a ser elaborado e ao público alvo. Cada sistema possui formas diferentes em disponibilizar o aplicativo em suas lojas (Stores), logo, é necessário planejar a montagem do app do curso tendo em vista em qual plataforma será submetido, qual o público que terá acesso e qual sistema será possível reproduzir o conteúdo desenvolvido.

4 METODOLOGIA E PESQUISA

A fim de alcançar o objetivo proposto por este estudo, foram realizadas buscas pelas melhores ferramentas e formatos multimídia que sejam compatíveis com todos os dispositivos mobiles atuais, independente da plataforma. Além disso, analisaram-se as metodologias modernas e tradicionais que melhor se adaptassem à realidade atual, de forma a se poder ser mais efetiva nos conteúdos gerados com a plataforma em desenvolvimento, concluindo-se pela MCE (Mapas de Conhecimento Estruturado), como afirmam FERREIRA et al. (2014, p. 514),

Os MCE tem o potencial de utilização em diversas áreas em que haja necessidade de transmissão de conhecimento e de sua retenção. Visualiza-se algumas delas na (i) educação, entre aluno e professor, contribuindo para o processo de aprendizagem; (ii) na comunicação, entre veículos e seus públicos, para retenção da informação;

Uma pesquisa teórica foi empreendida na busca de caminhos já traçados por outros pesquisadores na área do ensino. Posteriormente, a pesquisa exploratória foi efetuada a partir das provas de conceito e experimentações, novos métodos heurísticos foram acrescentados aos tecnicamente obtidos, qualificados e medidos, para se obter um maior domínio dos fenômenos e técnicas envolvidas nos processos focais do trabalho.

Este trabalho também contou com uma pesquisa experimental com diversas prototipações para prova de conceito na busca de se obter um domínio das habilidades para que se traduzissem em competências na efetivação dos objetivos. Pesquisas laboratoriais também foram empreendidas, através de testes e provas de conceitos, com todas as plataformas de SO existentes nas soluções mobiles, bem como em representantes de cada classe de dispositivos, para garantir que as autorias geradas rodassem corretamente em todos os dispositivos, o que é fundamental no processo de ensino.

Ainda foram realizadas pesquisas de campo antes, pelo que o usuário desejaría encontrar em uma autoria e a dinâmica de utilização e navegação, como uma complementar para testes após o protótipo funcionando. Para criação de um curso em plataformas mobiles foi desenvolvido uma biblioteca de fácil utilização por qualquer usuário disposto a desenvolver uma aplicação m-Learning, contendo todas as funções empacotadas por ações (uma ação por função), para que se possa montar uma autoria como se monta um produto em uma linha seriada.

Além de se gerar uma biblioteca para que outros projetistas possam elaborar o

próprio montador, com as particularidades que lhe são desejadas, foi proposto nesta pesquisa um montador de autoria, em que as ferramentas são compostas, também, pelos componentes desta biblioteca, com o perfil apresentado na introdução deste trabalho, ou seja, evitando sobrecarga cognitiva multimídia (as pessoas tendem a abusar dos recursos), buscando a degradação cognitiva focando em uma ação de cada vez, reduzindo-se o número de chunks de informações e promovendo uma coesão e coerência entre ações, páginas, conforme proposto nos MCE. A seguir, as funções dessa biblioteca e sua aplicabilidade serão ilustradas.

4.1 Biblioteca Educação a Distância (EAD)

A plataforma escolhida para o projeto foi o CORONA SDK, uma plataforma gratuita, multiplataforma e multidevice, inclusive para Windows Phone, Mac desktop, PC-Windows, Apple TV e Android TV.

O motivo, além dos descritos no parágrafo anterior, consiste, também, por ser, no mínimo, dez vezes mais rápido desenvolver um aplicativo em CORONA SDK, do que em outras plataformas ou linguagens, como Java, por exemplo. Outro destaque é que o código fica muito mais reduzido, cerca de 10 a 100 vezes menos linhas de programação do que Objective C e Java (<https://coronalabs.com/products/corona-is-10x-faster/>). Além disso, a plataforma utiliza a programação Lua, simples de aprender e de se fazer implementações, além de ser uma linguagem gratuita e 100% brasileira.

Neste trabalho, a biblioteca desenvolvida para utilização de componentes para instrumentação de cursos e aulas para educação a distância foi nomeada como **libEAD**. A forma de utilizá-la com a plataforma CORONA SDK é bastante simples, bastando importá-la em uma variável qualquer como, por exemplo, **ead**, conforme a seguir:

```
ead = require ("libEAD")
```

A biblioteca é composta pelas funções que disponibilizam a utilização dos recursos necessários para um e-book multimídia, conforme já descrito. Os parâmetros de cada função são array¹ de *chave* e *valores* definidos como **atributos**. Dessa forma, toda função **ead** recebe

¹ “Array” é uma estrutura de dados que armazena uma coleção de elementos, de tal forma que cada um possa ser identificado por, pelo menos, um índice ou uma chave. Essa estrutura de dados também é conhecida como variável indexada, vetor (para arranjos unidimensionais) e matriz (para arranjos bidimensionais).

como parâmetro um array **atributos**. A nomenclatura é composta da seguinte forma: *ead.[função](atributos)*, em que **atributos** é representado por *{chave=valor, ...}*.

4.1.1 Função Colocar Imagem

Função disponível para inserir uma imagem em qualquer tamanho e posição em uma tela. A imagem pode ser um hiperlink, desde que seja atribuído endereço do link em uma url. Esta função pode ser configurada com os seguintes atributos: arquivo, comprimento, altura, x, y, url, *posicao.

- **Posicao**: base (colocar a imagem na base da tela), centro (colocar a imagem no centro da tela) e topo (colocar a imagem no topo da tela).

Exemplo de uso da função:

```
ead.colocarImagen({arquivo="barra-superior.png", altura=200})
```

4.1.1.1 Codificação em CoronaSdk-Lua (biblioteca libEA.lua)

```

86   ----- COLOCAR IMAGEM -----
87   -- Funcao responsavel por colocar uma imagem
88   -- na tela, podendo ser tela inteira
89   -- ou dimensionada
90   --
91   -- ATRIBUTOS: arquivo,comprimento,altura,x,y,*posicao
92   -- *Posicao: base, centro, topo
93
94   local function colocarImagem(atributos)
95       if naoVazio(atributos) then
96           if(atributos.arquivo) then
97               local imagem = display.newImage(atributos.arquivo)
98
99               --POSICAO CENTRO
100              if(atributos.posicao=='centro') then
101                  imagem.x = display.contentCenterX
102                  imagem.y = display.contentCenterY
103
104              --POSICAO TOPO
105              elseif(atributos.posicao=='topo') then
106                  imagem.x = W - (W/2)
107
108              --POSICAO BASE
109              elseif(atributos.posicao=='base') then
110                  imagem.x = W - (W/2)
111                  imagem.y = H - (imagem.height/2)
112
113              else
114                  imagem.anchorX=0;
115                  imagem.anchorY=0;
116                  imagem.width=W;
117                  imagem.height=H;
118
119              end
120
121              if (atributos.comprimento) then
122                  imagem.width=atributos.comprimento;
123              end
124              if (atributos.altura) then
125                  imagem.height=atributos.altura
126              end
127              if (atributos.x) then
128                  imagem.x=atributos.x
129              end
130
131              if (atributos.y) then
132                  imagem.y=atributos.y
133              end
134
135      return imagem;
end
end
M.colocarImagem = colocarImagem

```

FIGURA 5 - Codificação Colocar Imagem (Fonte: Do Autor).

4.1.2 Função Colocar Som

Função disponível para colocar som em ambientes e objetos. Os arquivos de sons podem ser do tipo WAV e MP3.

Exemplo de uso da função:

```
ead. colocarSom({arquivo = "song.mp3" })
```

4.1.2.1 Codificação em CoronaSdk-Lua (biblioteca libEA.lua)

```

138      -----
139      -- COLOCAR SOM
140      -- Funcao responsavel por tocar som,
141      -- aceita arquivos wav e mp3
142      -- ATRIBUTOS: arquivos
143      -----
144      local function colocarSom(atributos)
145          media.stopSound()
146          if naoVazio(atributos) then
147              if(atributos.arquivo) then
148                  return media.playSound(atributos.arquivo)
149              end
150          end
151      end
152      M.colocarSom = colocarSom

```

FIGURA 6 - Codificação Colocar Som (Fonte: Do Autor).

4.1.3 Função Colocar Botão

Função disponível para criar um botão, que pode ser personalizado com os atributos, a seguir: fundo, título, altura, comprimento, x, y, *posicao, *acao.

Os campos *posicao* e *acao* aceitam parâmetros definidos.

- **Posicao:** base (coloca o botão na base do dispositivo), centro (coloca o botão no centro) e topo (coloca o botão no topo do dispositivo).

- **Acao:** *iniciaCurso* (inicia o curso a partir da primeira tela), *iniciaPagina* (inicia a tela corrente novamente).

Exemplo de uso da função:

```
ead. colocarBotao ({comprimento=300, titulo='oi sou um botão', fundo='fundo-verde.png'})
```

4.1.3.1 Codificação em CoronaSdk-Lua (biblioteca libEA.lua)

```

154      -- COLOCAR BOTAO
155      -- Funcao responsavel por criar um botao
156      -- de acordo com as opcoes desejadas e acoes pre-definidas
157      -- ATRIBUTOS: fundo,titulo,altura,comprimento,x,y,*posicao,*acao
158      -- *Posicao: base, centro, topo
159      -- *Acoes: iniciaCurso, iniciaPagina
160
161
162      local function colocarBotao(atributos)
163          -- DEFINE ATRIBUTOS DEFAULT
164          if atributos == nil then
165              atributos = {}
166          end
167          if atributos.fundo == nil then
168              atributos.fundo = "fundo-azul.png"
169          end
170          -- ACAO DO BOTAO
171          local function handleButtonEvent( event )
172              if ( "ended" == event.phase ) then
173                  if(atributosacao) then
174                      print( "Botao pressionado: "..atributosacao )
175                      verificaAcaoBotao(atributosacao)
176                  end
177              end
178          end
179
180          local botao = widget.newButton(
181          {
182              width = 240,
183              height = 120,
184              defaultFile = atributos.fundo,
185              label = "Botao",
186              onEvent = handleButtonEvent
187          }
188          --PARAMETRO TITULO
189          botao:setLabel( atributos.titulo )
190          --PARAMETRO FUNDO
191          botao.setDefaultFile =atributos.fundo
192          --PARAMETRO ALTURA
193          botao.height =atributos.altura;
194          --PARAMETRO COMPRIMENTO
195          botao.width =atributos.comprimento;
196          --POSICAO X, Y
197          if(atributos.x) then
198              botao.x = atributos.x;
199          end
200          if(atributos.y) then
201              botao.y = atributos.y;
202          end
203          --POSICAO CENTRO
204          if(atributos.posicao=='centro') then
205              botao.x = display.contentCenterX
206              botao.y = display.contentCenterY
207          end
208          --POSICAO TOPO
209          if(atributos.posicao=='topo') then
210              botao.x = W - (W/2)
211          end
212          --POSICAO BASE
213          if(atributos.posicao=='base') then
214              botao.x = W - (W/2)
215              botao.y = H - (botao.height/2)
216          end
217          return botao
218      end
219      M.colocarBotao = colocarBotao

```

FIGURA 7 - Codificação Colocar Botão (Fonte: Do Autor).

4.1.4 Função Colocar Vídeo

Função disponível para que um vídeo seja inserido. Os arquivos podem ser do tipo MOV, MP4 e M4V, ou o vídeo pode ser incorporado por um link. Pode ser personalizado com os seguintes atributos: arquivo, url.

Exemplo de uso da função:

```
ead.colocarVideo({arquivo="vídeo.mp4"})
```

4.1.4.1 Codificação em CoronaSdk-Lua (biblioteca libEA.lua)

```

221 -----
222 --- COLOCAR VIDEO
223 --- Funcao responsavel por tocar um Video
224 --- de acordo com arquivo: (mp4,mov, m4v)
225 --- ATRIBUTOS: arquivo
226 -----
227     local function colocarVideo(atributos)
228         if naoVazio(atributos) then
229             if(atributos.arquivo) then
230                 local onComplete = function( event )
231                     print( "Video finalizado!" )
232                 end
233             return media.playVideo( atributos.arquivo, true, onComplete )
234         end
235     end
236     M.colocarVideo = colocarVideo
237 
```

FIGURA 8 - Codificação Colocar Vídeo (Fonte: Do Autor).

4.1.5 Função Colocar Animação

Função disponível para gerar uma sequência de imagens, podendo ou não ter som. Este recurso gera uma animação por frames, de acordo com o tempo (em milissegundos) fornecido para cada imagem. Pode ser configurada com os seguintes atributos: imagens, tempoPorImagem, som.

Lista de imagens para sequência de animação:

```
imagensAnimacao = {"0.jpg","05.jpg", "010.jpg","020.jpg"}
```

Exemplo de uso da função:

```
ead.colocarAnimacao({apresentacao="central",
    imagens=imagensAnimacao, tempoPorImagen=350, som="raio.mp3"})
```

4.1.5.1 Codificação em CoronaSdk-Lua (biblioteca libEA.lua)

```
239 -----  
240 -- COLOCAR ANIMACAO  
241 -- Funcao responsavel por gerar uma animacao  
242 -- de acordo com as imagens, tempo (ms) e som.  
243 -- ATRIBUTOS: imagens,tempoPorImagen,som  
244 -----  
245 local function colocarAnimacao(atributos)  
246     local imagensAux = {}  
247     media.playSound(atributos.som)  
248     local imagens = atributos.imagens  
249     for i=1,#imagens do  
250         toqueHabilitado=false  
251         local listenerTimer = {}  
252         function listenerTimer:timer()  
253             print("Imagen animacao: "..imagens[i]);  
254             imagensAux[i] = display.newImage(imagens[i]);  
255             imagensAux[i].y = display.contentCenterY  
256             imagensAux[i].x = display.contentCenterX  
257             if(atributos.apresentacao=="cheia") then  
258                 imagensAux[i].y = 0  
259                 imagensAux[i].x = 0  
260                 imagensAux[i].anchorX = 0  
261                 imagensAux[i].anchorY = 0  
262                 imagensAux[i].width = W  
263                 imagensAux[i].height = H  
264             end  
265             if(atributos.apresentacao=="central") then  
266                 imagensAux[i].y = display.contentCenterY  
267                 imagensAux[i].x = display.contentCenterX  
268                 imagensAux[i].width = W  
269                 --imagensAux[i].height = H/2  
270             end  
271             if (i > 1) then  
272                 imagensAux[i-1]:removeSelf()  
273                 print("Imagen removida: "..imagens[i-1]);  
274             end  
275             if (i == #imagens) then  
276                 imagensAux[i]:removeSelf()  
277                 toqueHabilitado=true  
278             end  
279         end  
280         local tempoPorImagen = atributos.tempoPorImagen  
281         timerAnimacao = timer.performWithDelay(i *tempoPorImagen, listenerTimer,1)  
282     end  
283     print("Ultima Imagem "..imagens[#imagens]);  
284     return timerAnimacao  
285 end  
M.colocarAnimacao = colocarAnimacao
```

FIGURA 9 - Codificação Colocar Animação (Fonte: Do Autor).

4.1.6 Função Colocar Plano de Fundo

Função disponível para colocar o mesmo plano de fundo em todas as telas, conforme uma cor pré-definida (branco, azul, preto, vermelho e verde) ou um arquivo de imagem para tela inteira. Pode ser configurada com os seguintes atributos: cor, arquivo.

Exemplo de uso da função:

```
ead. colocarPlanoDeFundo({cor="branco"})
```

4.1.6.1 Codificação em CoronaSdk-Lua (biblioteca libEA.lua)

```

288   -----
289   -- COLOCAR PLANO DE FUNDO
290   -- Funcao responsavel por colocar um plano de fundo em todas as telas.
291   -- ATRIBUTOS: cor, arquivo
292   --
293   -----
294   local function colocarPlanoDeFundo(atributos)
295     if naoVazio(atributos) then
296       if (atributos.cor) then
297         if(atributos.cor=='branco') then
298           display.setDefault( "background", 1, 1, 1 )
299         end
300         if(atributos.cor=='preto') then
301           display.setDefault( "background", 0, 0, 0 )
302         end
303         if(atributos.cor=='azul') then
304           display.setDefault( "background", 0, 0, 1 )
305         end
306         if(atributos.cor=='vermelho') then
307           display.setDefault( "background", 1, 0, 0 )
308         end
309         if(atributos.cor=='verde') then
310           display.setDefault( "background", 0, 1, 0 )
311         end
312       end
313     end
314
315     if(atributos.arquivo) then
316       imagemFundo = display.newImage(atributos.arquivo)
317       imagemFundo.anchorX=0;
318       imagemFundo.anchorY=0;
319       imagemFundo.width=W;
320       imagemFundo.height=H;
321       elseif (imagemFundo) then
322         imagemFundo:removeSelf()
323       end
324     end
325   end
326   M.colocarPlanoDeFundo = colocarPlanoDeFundo

```

FIGURA 10 - Codificação Colocar Plano de Fundo (Fonte: Do Autor).

4.1.7 Função Colocar Texto

Função disponível para introduzir um texto. Ele pode ser um *hyperlink*, caso seja, deve ser fornecido uma URL. Pode ser configurado com os seguintes atributos: texto, url, tamanho, eNegrito, *fonte, *posicao.

Em que o campo *fonte* e *posicao* aceita parâmetros definidos:

- **Fonte**: arial, times e calibri.
- **Posicao**: topo, centro e base.

Exemplo de uso da função:

```
ead. colocarTexto ({tamanho=20, texto='Texto de exemplo para a função  
colocarTexto.', eNegrito='sim', url='http://google.com'})
```

Obs.: Vários textos podem ser colocados em sequência. Caso seja necessário saltar linha, utilizar o “\n” entre um texto e outro.

4.1.7.1 Codificação em CoronaSdk-Lua (biblioteca libEA.lua)

```

226  -- COLOCAR TEXTO
227  -- Função responsável por criar um texto
228  -- de acordo com as opções desejadas e ações pre-definidas
229  -- ATRIBUTOS: texto, url, tamanho, eNegrito,*posicao,*fonte
230  -- *Posicao: topo, centro, base
231  -- *Fonte: arial, times, calibri
232
233
234 local function colocarTexto(atributos)
235     if naoVazio(atributos) then
236         local options =
237         {
238             text = "",
239             x = (W/2)+20,
240             y = M/2,
241             width = W-20,
242             height = 0,
243             font = native.systemFont,
244             fontSize = 20
245         }
246         --POSICAO CENTRO
247         if(atributos.posicao=='centro') then
248             options.x = display.contentCenterX
249             options.y = display.contentCenterY
250         --POSICAO TOP
251         elseif(atributos.posicao=='top') then
252             options.x = W - (W/2)
253             options.y = display.contentCenterY - M/3
254         --POSICAO BASE
255         elseif(atributos.posicao=='base') then
256             options.x = W - (W/2)
257             options.y = M - display.contentCenterY/3
258         end
259         --FONTES
260         if(atributos.fonte =='arial') then
261             options.font = 'Arial'
262         elseif(atributos.fonte=='times') then
263             options.font = 'Times'
264         elseif(atributos.fonte=='calibri') then
265             options.font = native.systemFont
266         end
267
268         if(atributos.texto) then
269             options.text = atributos.texto;
270         end
271
272         if(atributos.tamanho) then
273             options.fontSize = atributos.tamanho;
274         end
275         if(atributos.eNegrito and atributos.eNegrito=='sim') then
276             options.font = native.systemFontBold;
277         end
278         local texto = display.newText( options )
279         texto:setFillColor( 0, 0.2, 0.13 ) --verde escuro
280         if(atributos.url) then
281             local function onClickLink(event)
282                 system.openURL(atributos.url)
283             end
284             texto:addEventListener('touch', onClickLink)
285         end
286         return texto
287     end
288 end

```

FIGURA 11 - Codificação Colocar Texto (Fonte: Do Autor).

4.1.8 Função Colocar Questão

Função disponível para colocar uma questão com enunciado e alternativas, no máximo, cinco respostas, em que serão disponibilizadas aleatoriamente. É possível inserir justificativas para as respostas incorretas, estas respostas são apresentadas, junto às justificativas, quando solicitadas pelo usuário. Existem duas opções de questões, a opção com uma alternativa correta ou assinaladas como V (verdadeiro) ou F (falso). Esta função pode ser

personalizada com os seguintes atributos: enunciado, respostasCorretas, respostasIncorrectas, justificativas, *tipo.

Em que o campo *tipo* aceita parâmetros definidos:

- **Tipo**: vf (Questão onde as respostas podem ser verdadeiras ou falsas), umaEscolha (Questão onde deve haver uma resposta correta).

Lista de respostas corretas, incorretas e justificativas, para colocar na questão:

```
respostasCorretas = {"Brasil Foi descoberto por Pedro Alves Cabral.",  
"Foi descoberto por volta do ano de 1500."}
```

```
respostasIncorrectas = {"Brasil Foi descoberto por Cristóvão Colombo.",  
"O Brasil foi descoberto por espanhóis e ingleses. "}
```

Exemplo de uso da função:

```
ead. colocarQuestao ({tipo='V/F', respostasCorretas= respostasCorretas,  
respostasCorretas= respostasIncorrectas, enunciado='Assinale V para  
verdadeiro ou F para falso: ', justificativas= justificativas })
```

```
justificativas = {"O Brasil foi descoberto por Pedro Alves Cabral por  
volta de 1500.", "O Brasil foi descoberto por caravelas portuguesas."}
```

respostasCorretas: É uma lista de respostas consideradas certas, entre aspas simples “ ” e separadas por “,”. Ex.: {‘Respostas certa 1’, ‘Resposta certa 2’, ‘Resposta certa 3’}

respostasIncorrectas: É uma lista de respostas consideradas certas, entre aspas simples “ ” e separadas por “,”. Ex.: {‘Respostas errada 1’, ‘Resposta errada 2’, ‘Resposta errada 3’}

Caso seja definido o tipo **vf**, as respostas corretas serão consideradas verdadeiras e as incorretas, falsas. Já para o tipo **umaEscolha**, as respostas corretas serão a opção correta e as incorretas as opções erradas.

Quando a opção verificar respostas é selecionado, na tela será apresentada a respostas incorretas e suas respectivas justificativas, desde que seja definida a lista de **justificativas**. Esta lista deve ser na mesma ordem da lista de **respostasIncorretas**.

4.1.8.1 Codificação em CoronaSdk-Lua (biblioteca libEA.lua)

```

298  -- COLOCAR QUESTAO
299  -- Funcao responsavel por criar questoes
300  -- de acordo com as opcoes desejadas e acoes pre-definidas
301  -- ATRIBUTOS: enunciado, respostasCorretas, respostasIncorretas,
302  -- justificativas, *tipo.
303  -- *Tipo: V/F, umaEscolha
304  --
305
306 local function colocarQuestao(atributos)
307     local enunciado;
308     --Posicao incialis do questao
309     local inicioX = 0;
310     local inicioY = 0;
311     local respostasFalsas = {};
312     if naoVazio(atributos) then
313         if(atributos.enunciado) then
314             enunciado = display.newText(atributos.enunciado, inicioX, inicioY, (W), 0, native.systemFontBold,36)
315             enunciado:setFillColor( 0, 0.2, 0.13 )
316             enunciado.anchorX = 0 ; enunciado.x=20; enunciado.y=H/4
317             inicioX = enunciado.x; inicioY = enunciado.y
318             --enunciado.anchorY = 200;
319         end
320         local questoes = {};
321         local respostas = {};
322         if(atributos.tipo) then
323             if (atributos.respostasCorretas) then
324                 respostas = juntarListas(respostas,atributos.respostasCorretas)
325             end
326             if (atributos.respostasIncorretas) then
327                 respostasFalsas = atributos.respostasIncorretas
328                 respostas = juntarListas(respostas, atributos.respostasIncorretas)
329             end
330             for i=1,#respostas do
331                 print(respostas[i])
332                 local textoResp = display.newText(i..".(" ..respostas[i], inicioX+20, inicioY+(80*i), W-30, 0, native.systemFont,32)
333                 textoResp:setFillColor( 0, 0.2, 0.13 )
334                 textoResp.anchorX = 0
335                 questoes[i] = textoResp
336             end
337
338             local justificativas = {"N/D"};
339             local textoRespostas = "";
340             if (atributos.justificativas and #atributos.justificativas == #respostasFalsas) then
341                 justificativas = atributos.justificativas
342             end
343
344             -- Verificar resposta(s) incorreta(s)
345             local function vefificarQuestoes(event)
346
347                 if "ended" == event.phase then
348                     if (atributos.tipo =='V/F') then
349                         local txtFalsas = display.newText("A(s) resposta(s) falsa(s): ",50, H-450,W-30,0, native.systemFontBold, 28 );
350                         txtFalsas:setFillColor( 1, 0, 0 )
351                         txtFalsas.anchorX = 0;
352                         for i=1,#respostasFalsas do
353                             textoRespostas = "- "..respostasFalsas[i];
354                             if (justificativas and #justificativas > 0) then
355                                 textoRespostas = textoRespostas.." (Justificativa: "..justificativas[i]..")";
356                             end
357                             txtRespQuestoes = display.newText(textoRespostas,50, (H-450)+(i*82), W-60,0, native.systemFont, 25 );
358                             txtRespQuestoes:setFillColor( 1, 0, 0 )
359                             txtRespQuestoes.anchorX = 0
360                         end
361                     elseif (atributos.tipo =='umaEscolha') then
362                         textoRespostas = "A resposta é: "..respostasFalsas[1].." (Justificativa: "..justificativas[1]..")";
363                         textoRespostas = display.newText( textoRespostas,50, H-370,W-30,0, native.systemFont, 25 );
364                         textoRespostas:setFillColor( 1, 0, 0 )
365                         textoRespostas.anchorX = 0
366                     end
367                 end
368             end
369             local textoVerificaQuestoes = display.newText( "Clique para verificar questoes...", 60, H-70, native.systemFont, 36 )
370             textoVerificaQuestoes.anchorX=0
371             textoVerificaQuestoes:setFillColor( 0, 0.2, 0.13 )
372             textoVerificaQuestoes:addEventListener('touch', vefificarQuestoes)
373         end
374     return enunciado, questoes
375 end
376 end

```

FIGURA 12 - Codificação Colocar Questão (Fonte: Do Autor).

4.1.9 Criar Curso

Função disponível para criar o Curso personalizado, de acordo com as telas fornecidas. Além das telas, recebe como atributo a página em que será apresentada a tela, segundo o índice que coincide com o número da página. Pode ser configurada com os seguintes atributos: telas[imagem, imagens, botao, som, video, animacao], pagina.

Segue a função:

```
ead. criarCurso (telas, 1)
```

De acordo com os atributos que são definidos no atributo *telas* são chamadas as funções definidas acima. Por exemplo, se uma *tela* conter *botão* e *imagem*, em que os atributos para o botão são `{comprimento=300, fundo="verde"}` e para a imagem `{comprimento=300, fundo="verde"}`, ficaria da seguinte forma:

```
tela={botao={comprimento=300, fundo="verde"},  
{ imagem={comprimento=300, fundo="verde"}}}
```

Assim, serão chamada as funções **ead.colocarBotao(atributos)** e **ead.colocarImagen(atributos)**.

Dessa forma, criando outra tela com os mesmo atributos:

```
tela2={botao={comprimento=300,fundo="verde"},  
{imagem={comprimento=300, fundo="verde"}}}
```

O objeto **telas** seria igual a `{ tela , tela2 }`.

4.1.9.1 Codificação em CoronaSdk-Lua (biblioteca libEA.lua)

```

12   -- CRIAR CURSO
13   -- Função responsável por criar o Curso personalizado
14   -- de acordo com as telas fornecidas e a página que
15   -- apresentaria a tela na ordem do valor da página
16   -- ATRIBUTOS: telas[imagem,imagens,botao,som,video,animacao],pagina
17   --
18   -----
19   local function criarCurso(telas, pagina)
20       local i = pagina;
21       print("TELAS: "..i)
22       --remover atributos anteriores
23       media.stopSound()
24       if(imagem ~= nil) then
25           imagem:removeSelf()
26       end
27       if(imagens ~= nil) then
28           for ims=1, #imagens do
29               imagens[ims]:removeSelf()
30           end
31           imagens = {}
32       end
33       if(botao ~= nil) then
34           botao:removeSelf()
35       end
36
37       --verifica se tem imagem
38       if(telas[i].imagem) then
39           imagem = ead.colocarImagem(telas[i].imagem)
40       end
41       if(telas[i].imagens) then
42           imagens = {}
43           for ims=1, #telas[i].imagens do
44               imagens[ims] = ead.colocarImagem(telas[i].imagens[ims])
45           end
46       end
47       --verifica se tem som
48       if(telas[i].som) then
49           som = ead.colocarSom(telas[i].som)
50       end
51       --verifica se tem video
52       if(telas[i].video) then
53           video = ead.colocarVideo(telas[i].video)
54       end
55       --verifica se tem botao
56       if(telas[i].botao) then
57           botao = ead.colocarBotao(telas[i].botao)
58       end
59       --Verifica se tem animacao
60       if(telas[i].animacao) then
61           animacao = ead.colocarAnimacao(telas[i].animacao)
62       end
63   end
64   M.criarCurso = criarCurso

```

FIGURA 13 - Codificação Criar Curso (Fonte: Do Autor).

4.1.10 Função Criar Paginação

Função disponível para criar a navegação entre telas em forma de páginas. As telas podem ser transitadas para direita ou para esquerda. A interface de paginação fica disponível em todas as telas e não é permitido alternar de páginas, enquanto houver uma animação em andamento.

Segue a função:

ead. criarPaginacao ()

4.1.10.1 Codificação em CoronaSdk-Lua (biblioteca libEA.lua)

```

329 -----  

330 -- CRIAR PAGINACAO  

331 -- Cria a paginacao para alteracao de telas para direita ou --  

332 -- para esquerda. A interface de paginacao fica disponivel em --  

333 -- todas telas. E nao é permitido alternar de paginas enquanto --  

334 -- houver uma animacao em andamento.  

335 -----  

336  

337 local function criarPaginacao()  

338     Interface = display.newGroup()  

339  

340     local botaoEsquerdo = display.newRect(0,0,W/4,H)  

341     botaoEsquerdo:setFillColor(255,0,0)  

342     botaoEsquerdo.anchorX=0  

343     botaoEsquerdo.anchorY=0  

344     botaoEsquerdo.isVisible = false  

345     botaoEsquerdo.isHitTestable = true  

346     botaoEsquerdo.alpha = 0.3  

347     --Interface:insert(botaoEsquerdo)  

348  

349     local botaoDireito = display.newRect(0,0,W/4,H)  

350     botaoDireito:setFillColor(255,0,0)  

351     botaoDireito.x = W - (botaoDireito.width/2); botaoDireito.y = H/2;  

352     botaoDireito.isVisible = false  

353     botaoDireito.isHitTestable = true  

354     botaoDireito.alpha = 0.3  

355     --Interface:insert(botaoDireito)  

356  

357     local function passarPaginaDireita(e)  

358         print("PAGINA: "..paginaAtual)  

359         if(paginaAtual<#telas and toqueHabilitado) then  

360             paginaAtual=paginaAtual+1  

361             print("foi para: "..paginaAtual.." de "..#telas)  

362             criarCurso(telas, paginaAtual)  

363         end  

364     end  

365     --M.passarPaginaDireita = passarPaginaDireita  

366  

367     local function passarPaginaEsquerda(e)  

368         print("PAGINA: "..paginaAtual)  

369         if(paginaAtual<= #telas and paginaAtual>1 and toqueHabilitado) then  

370             paginaAtual=paginaAtual-1  

371             print("voltou para: "..paginaAtual.." de "..#telas)  

372             criarCurso(telas, paginaAtual)  

373         end  

374     end  

375     --M.passarPaginaEsquerda = passarPaginaEsquerda  

376  

377     --Define Pagina Atual  

378     paginaAtual=1  

379     botaoDireito:addEventListener("tap", passarPaginaDireita)  

380     botaoEsquerdo:addEventListener("tap", passarPaginaEsquerda)  

381  

382     M.criarPaginacao = criarPaginacao

```

FIGURA 14 - Codificação Criar Paginação (Fonte: Do Autor).

Esta função deve ser definida no início da aplicação, antes da criação de telas. Dessa forma, a interface fica disponível para toda aplicação.

4.1.11 Funções Auxiliares, Cabeçalho da Biblioteca e Chamada da Biblioteca no Programa Principal

Funções utilizadas pelas outras funções já descritas e necessárias para que o aplicativo seja gerado corretamente.

4.1.11.1 Funções Auxiliares

```

67
68  --  FUNCOES AUXILIARES
69  -----
70  local function naoVazio(s)
71      return s ~= nil or s ~= ''
72  end
73
74  local function verificaAcaoBotao( acao )
75      --if naoVazio(acao) then
76      if (acao=="iniciaPagina" and toqueHabilitado) then
77          criarCurso(telas, paginaAtual)
78      end
79      if (acao=="iniciaCurso" and toqueHabilitado) then
80          criarCurso(telas, 1)
81          paginaAtual=1
82      end
83  --end
84 end

```

FIGURA 15 - Codificação Funções Auxiliares (Fonte: Do Autor).

4.1.11.2 Cabeçalho da Biblioteca

```

1  --IMPORTS
2  local widget = require "widget";
3
4  local W = display.contentWidth;
5  local H = display.contentHeight;
6
7
8  local M = {}
9  local paginaAtual = 1
10 local toqueHabilitado = true

```

FIGURA 16 - Codificação Cabeçalho Biblioteca (Fonte: Do Autor).

4.1.11.3 Chamada da Biblioteca no Programa Principal a ser criado (main.lua)

```

--IMPORT LIBS
ead = require ("libEA")

```

FIGURA 17 - Codificação Chamada Biblioteca (Fonte: Do Autor).

5 MONTADORES

Seguindo a ideia da geração de um montador de autoria, que tutele de forma transparente ao conteudista, ofereceu-se uma sequência de ações de como preparar um curso que concretize os objetivos de aprendizagem traçados inicialmente, evitando resultados arbitrários, ambíguos, ou seja, os resultados matematicamente ortogonais, conforme explicados anteriormente.

O objetivo aqui é apresentar a metodologia de educação definida em uma prototipação de um motor para cursos em m-Learning ou AVA. O montador de autoria habilita o tutor a elaborar um curso que seja personalizável, desde a introdução à escolha da metodologia, através de mídias, imagens, áudio, textos, links e exercícios didáticos. A personalização parte de se já ter um dispositivo com o aplicativo instalado, contendo todas as opções apresentadas para criação de um curso didático em que serão utilizados os conceitos de aprendizagem aqui demonstrados.

Será apresentado em etapas, e através dos MCE aplicados, cada ferramenta com todas as personalizações específicas para a criação de cada tela, em que uma única tela será possível utilizar uma única ferramenta, corroborando com o Princípio da Navalha de Ockham (*Lexi Parsinoniae*) (OCKHAM, 1999). Cada tela deve assumir estritamente a informação necessária à explicação do mesmo, sem comprometer o entendimento do indivíduo para o objetivo da informação naquela tela.

O montador proposto declara uma barra inferior na tela do dispositivo, em que é possível selecionar a ferramenta desejada com suas opções de personalização para criação de métodos distintos para o conteúdo a ser aprendido. Quando o autor finalizar a tela, segue-se para a tela subsequente para elaboração de outro conteúdo. Toda etapa pode ser pré-visualizada para simular o que foi montado e, após iniciado, retorna-se para qualquer etapa (tela) criada e finaliza-se a autoria.

5.1 Etapa inicial do montador de autoria

A montagem de um curso inicia-se pelo o que será apresentado, ou seja, sua capa. A tela inicial é esta capa, com um título e o layout que definirá a identidade do que será aprendido. Dessa forma, pode ser feito a personalização com um plano de fundo de todas as telas para aquele curso.



FIGURA 18 - Ferramenta Inicial Montador (Fonte: Do Autor).

A barra inferior identifica as opções de personalização para o conteúdo de apresentação do curso. O processo de criação é dinâmico e, conforme as ferramentas sejam selecionadas, novas opções de criação que melhor atendam o desejo do educador na criação do seu curso, podem ser abertas.

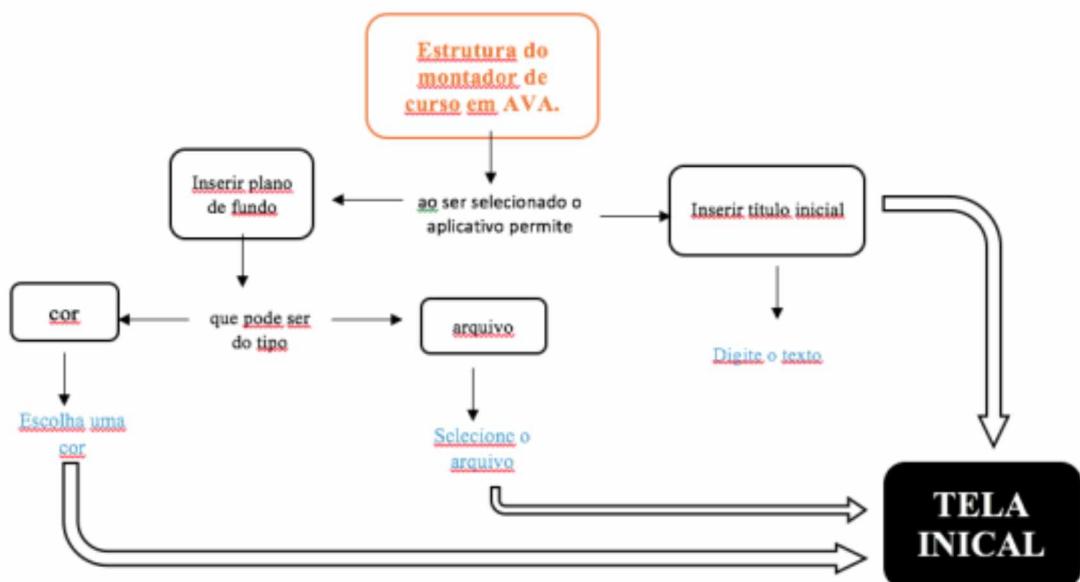


FIGURA 19 – MCE Estrutura do Montador de Curso (Fonte: Do Autor).

Após configuração da tela inicial e do plano de fundo para todas as outras telas, prossegue-se para a elaboração das telas de conteúdo de aprendizagem. Um novo menu de

ferramentas na barra inferior é apresentado com variáveis distintas a serem utilizadas para construção da informação a ser proposta.

5.2 Apresentação de textos

Os textos podem ser inseridos em uma página do curso, com opções de serem com Hiperlink, normal ou em formas de questão.



FIGURA 20 – Ferramenta Colocar Texto (Fonte: Do Autor).

Textos convencionais são inseridos apenas escolhendo a opção “**Normal**”. Caso seja de interesse do educador, podem ser adicionados outros textos, tais como:

- 1- posição **Início** (seria o primeiro texto, acima de todos os outros),
- 2- posição **Antes** do primeiro texto,
- 3- posição **Após** o último texto.

Outra forma de inserir um texto é como “**Hiperlink**”. Após o texto inserido, deve-se digitar a url desejada para ser aberto na web, quando clicado no link.



FIGURA 21 - Ferramenta Colocar Questão (Fonte: Do Autor).

Outra opção é a “Questão”, em que se escolhe o tipo, podendo ser **V** (verdadeiro) ou **F** (falso) ou de **escolha da alternativa correta**. Para tanto, entra-se com o título da Questão, as alternativas corretas e alternativas incorretas. Além disso, existe a opção de inserir justificativas para as alternativas incorretas, respectivamente, na mesma ordem destas alternativas.

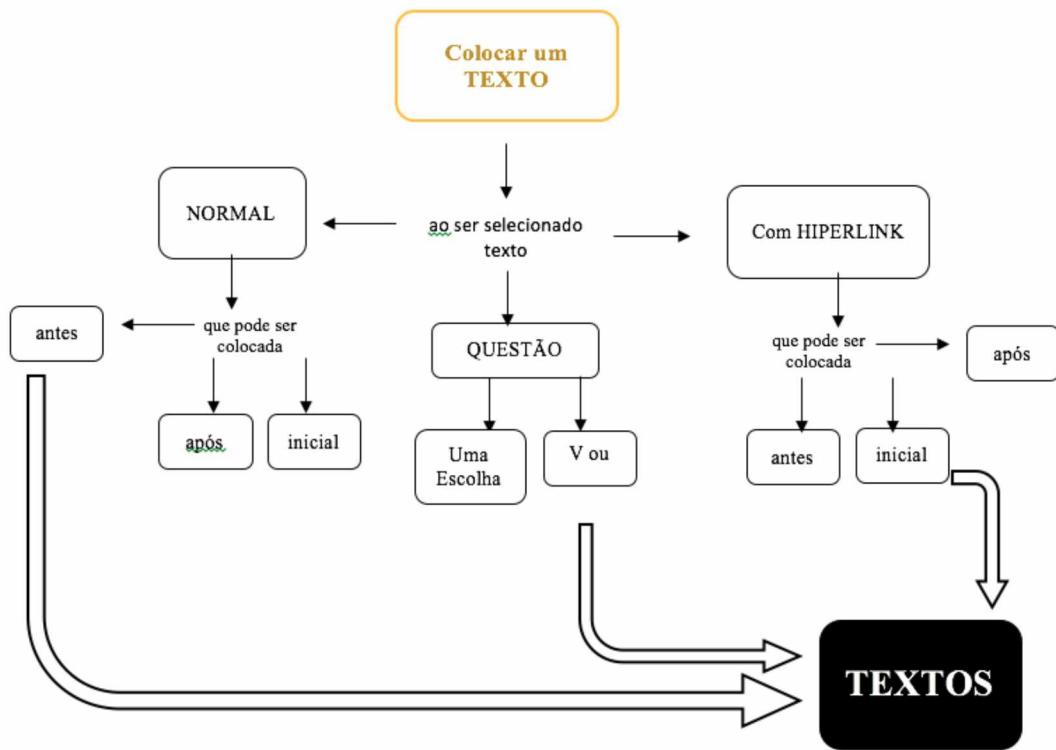


FIGURA 22 – MCE Colocar um Texto (Fonte: Do Autor).

5.3 Inserir imagem

Imagens nos formatos bmp, jpeg, jpg e png, podem ser inseridas em uma página do curso, desde que já esteja disponível para acesso no dispositivo.



FIGURA 23 - Ferramenta Colocar Imagem (Fonte: Do Autor).

A imagem escolhida pode ser posicionada:

- 1- no “**Centro**” (centro do dispositivo),
- 2- no “**Topo**” (parte superior),
- 3- na “**Base**” (parte inferior).

A imagem será dimensionada proporcionalmente de acordo com a altura e largura do dispositivo.

Por exemplo: caso a imagem seja do tipo paisagem (horizontalmente maior que na vertical), será dimensionada horizontalmente e na dimensão vertical irá diminuir para manter a resolução da figura, o mesmo acontecerá caso seja verticalmente maior.

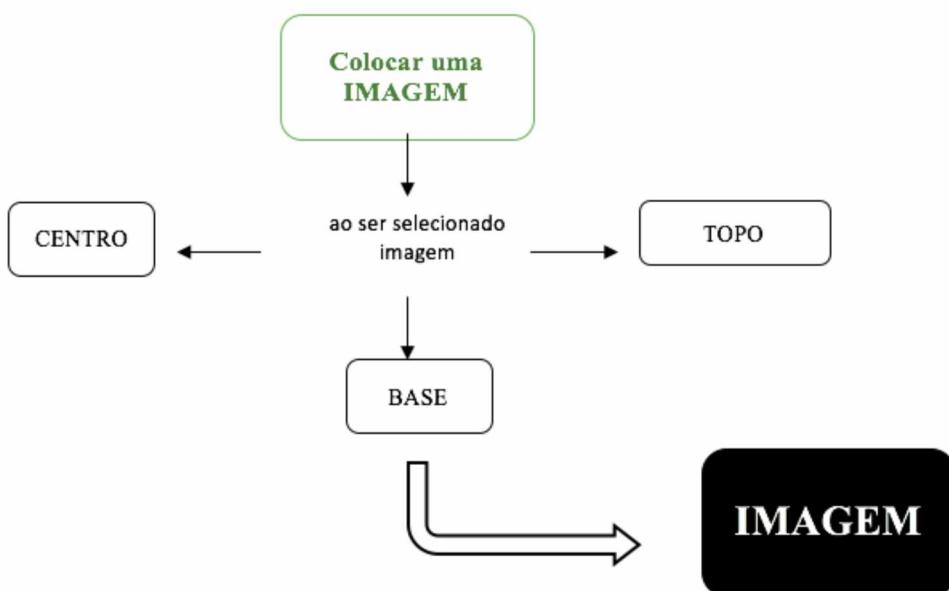


FIGURA 24 – MCE Colocar uma Imagem (Fonte: Do Autor).

5.4 Colocar som

Para inserir um som em uma página, bastam escolher um arquivo nos formatos mp3 ou WAV, disponíveis no dispositivo.



FIGURA 25 - Ferramenta Colocar Som (Fonte: Do Autor).

O áudio pode ser colocado junto a uma imagem ou uma animação. Quando a página que contém o áudio é alterada, este é interrompido. Caso seja retornado para a página com o som, o áudio será reproduzido do início. Quando adicionado um som, uma barra de menu, localizada na parte inferior do dispositivo é apresentada com as opções de PLAY, PAUSE e REPETIR.

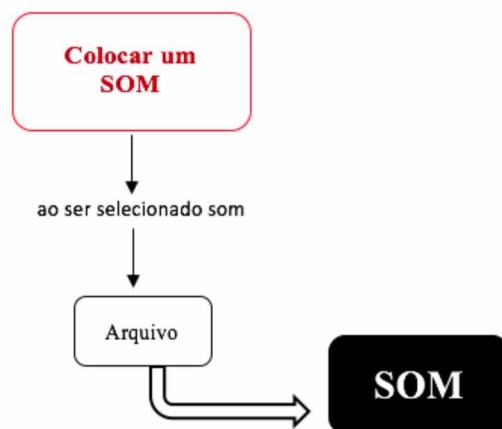


FIGURA 26 – MCE Colocar um Som (Fonte: Do Autor).

5.5 Inserir vídeo

Inserir um vídeo segue a mesma ideia de uma imagem, entretanto, vídeos podem ser inseridos a partir de um arquivo disponível no dispositivo ou incorporados, através de um endereço web (URL) de um vídeo no site Youtube.



FIGURA 27 - Ferramenta Colocar Vídeo (Fonte: Do Autor).

O vídeo inserido ou incorporado, assim como a imagem, é dimensionado proporcionalmente tanto na vertical quanto na horizontal.



FIGURA 28 – MCE Colocar um Vídeo (Fonte: Do Autor).

5.6 Gerar Animação

Uma animação é gerada por uma sequência de imagens alternadas em um pequeno intervalo de tempo. Dessa forma, é necessário escolher mais de uma imagem no dispositivo, definir um intervalo, em segundos, entre cada imagem e, se for desejado, escolher um arquivo de som.



FIGURA 29 - Ferramenta Colocar Animação (Fonte: Do Autor).

As imagens são posicionadas no centro do dispositivo, proporcionalmente, ao seu tamanho, sendo alternadas de acordo com o tempo definido. Caso este tempo não seja escolhido, o intervalo padronizado será de cinco segundos entre cada imagem. O som será exibido ao iniciar a página e finalizado de acordo com sua duração ou quando a página for alterada.

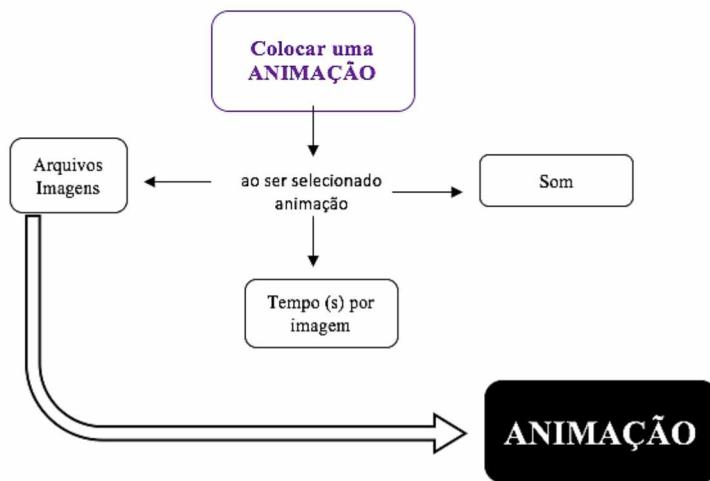


FIGURA 30 – MCE Colocar Animação (Fonte: Do Autor).

5.7 Finalizar e pré-visualizar o trabalho

O Corona SDK possui um simulador adaptável ao dispositivo que rodará o aplicativo, independente da plataforma utilizada (iOS, Android, Windows, ...).

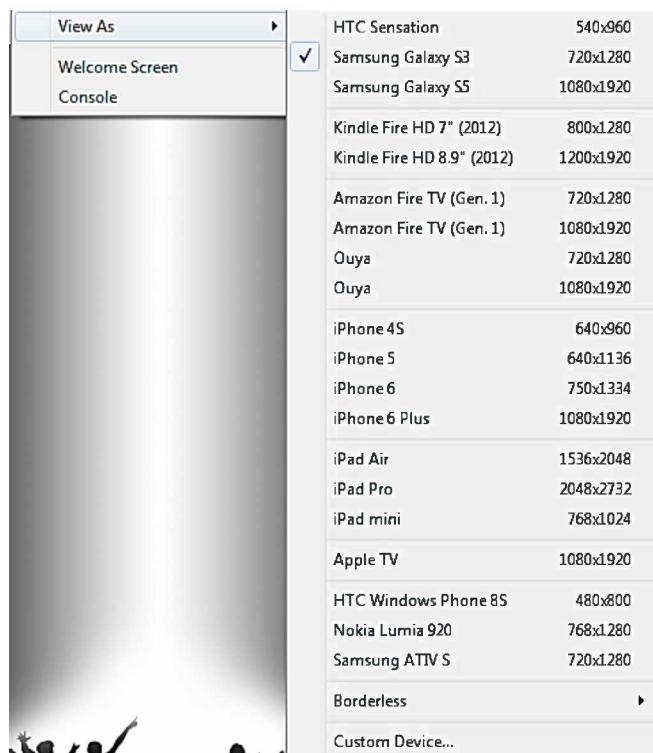


FIGURA 31 – Dispositivos Mobiles Corona SDK (Fonte: Do Autor).

O que a simulação mostrará será exatamente o que o aplicativo apresentará quando rodar. A compilação para qualquer destas plataformas é simples, bastando no menu do simulador solicitar a compilação para a plataforma em questão, o que é denominado de **build**.

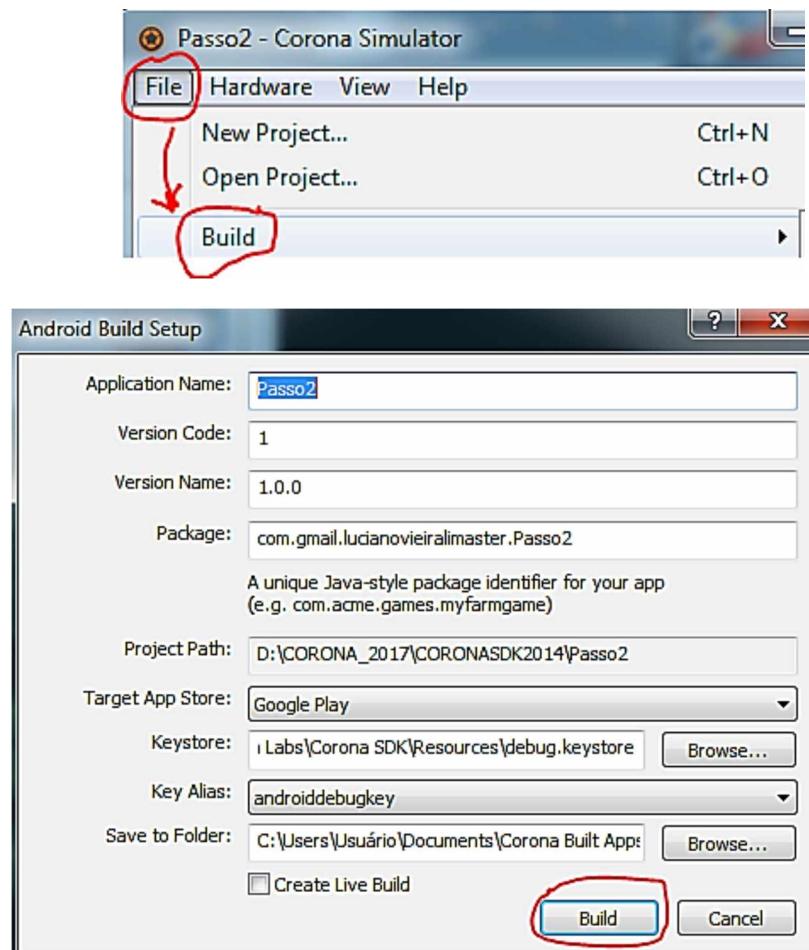


FIGURA 32 – Build Aplicação Corona SDK (Fonte: Do Autor).

6 CONCLUSÃO E RELEVÂNCIA

Conforme já apresentado, conteúdos voltados ao ensino, para competir e dividir atenção com os nativos efetivamente digitais (os que já nasceram tendo como primeira babá um tablet) devem possuir os mesmos recursos com que o aluno está acostumado a utilizar no cotidiano, sem obrigar a adotar novas (ou antigas) abstrações alienígenas ao mesmo.

Segundo Mayer (2001) e os MCE (Mapas de Conhecimento Estruturado), em aplicações multimídias, normalmente, fazem-se uso de recursos que utilizam mais de um sentido de percepção ao mesmo tempo, como, visão e audição, gerando sobrecarga cognitiva que pode levar a um desfoque, uma perda da informação que se deseja transmitir, já que o cérebro captura 70% da informação pela visão e 15% pela audição (o restante pelos outros sentidos, quando utilizados). Como o cérebro não é multitarefa, um sentido concorre com o outro na aquisição, e, desta forma, os dois são prejudicados, não capturando totalmente nem uma nem outra informação. Uma técnica para melhorar, seria que o áudio e a imagem retratassem a mesma coisa, mas, como nem sempre (ou quase nunca) acontece, é melhor evitar. Este trabalho visou eliminar esta concorrência de informações, uma sobrecarga, em que o educador pode impor, ou privilegiar, o uso de um único e mais adequado sentido na aquisição da informação desejada.

Observa-se, portanto, que cada vez mais a mobilidade vem tomando espaço em todo o mundo, assim como nos processos de aprendizagem. Os aprendizes, principalmente, os nativos efetivamente digitais (até 26 anos de idade) utilizam o celular para todas as tarefas relacionadas a conhecimento e relacionamentos (redes sociais), transcendendo para comércio e até transações financeiras. Assim, o acesso a cursos, principalmente, virtuais, bem como em tarefas e distribuição de conteúdo de cursos presenciais, vêm sendo realizados de forma mobile, o que se denomina de m-Learning.

Este trabalho acrescenta a este estado da arte, a possibilidade de que não apenas se acesse uma autoria, mas que se possa implementá-la e efetivá-la, através do celular ou tablet. A independência de se ter um computador com acesso à nuvem (o que implica em se ter um sinal com largura de banda, ou velocidade, adequada), resulta numa maior autonomia tanto ao conteudista quanto ao gerador de materiais de estudo e de apoio.

Para tanto, a construção do motor gerador do aplicativo de autoria seguiu os fundamentos de se evitar sobrecarga cognitiva, de inserir a degradação cognitiva (evitar que informações indesejadas ganhem foco no processo de aprendizagem), provocando e evocando uma melhor participação do aprendiz com um foco centrado em um sentido de cada vez, o

que maximiza o aprendizado, conforme orienta e comprova ao MCE (Mapas de Conhecimento Estruturado). Enfim, os resultados obtidos comprovam e cumprem os objetivos traçados para este trabalho.

6.1 Trabalhos futuros

Este trabalho, com a geração de uma biblioteca com um motor de efetivação eficiente para conteúdo multimídia, abre também espaço e potencialidade para que outros profissionais possam gerar Montadores de Autoria personalizados, sem perder o foco de se degradar o indesejado (reduzindo os chunks de informação) e dar foco no que se deseja transmitir de conhecimento, evitando a sobrecarga cognitiva.

Assim, o foco principal dos trabalhos futuros se dá no desenvolvimento de Montadores de Autoria personalizados e adaptados a diversas áreas do domínio do conhecimento, não só o cognitivo, sempre tendo em mente em também fazer um espelho, na nuvem, do conteúdo gerado. Desta forma, os educadores poderão alterar os conteúdos dos aplicativos à distância, atualizando o conteúdo já gerado, eliminando e acrescentando novos itens, tornando a autoria mais dinâmica e adaptativa.

Foi proposto e evidenciado, através dos estudos citados, que poupar sobrecarga cognitiva multimídia (como as pessoas tendem a abusar dos recursos), através do montador de autoria com a representação na introdução desta dissertação, ou seja, buscando a degradação cognitiva focando em uma ação de cada vez, reduz-se o número de chunks de informações, promovendo uma coesão e coerência na transmissão do conhecimento, conforme proposto nos MCE.

O motor implementado neste trabalho, conclui-se, é totalmente aderente ao processo biológico ao aprendizado (STRAYER et al., 2013) e aos anseios do confeudista na geração de material, principalmente, para os nativos efetivamente digitais.

REFERÊNCIAS

ALMEIDA, M. E. B. de. Educação à distância na internet: abordagens e contribuições dos ambientes digitais de aprendizagem. **Educação e Pesquisa**, São Paulo, v. 29, n. 2, p. 327-340, dez. 2003. Disponível em: <<http://dx.doi.org/10.1590/S1517-97022003000200010>>. Acesso em: 25 jan. 2017.

AMAZON. **Amazon: Get to Know Us**. Disponível em: <<https://www.amazon.com/>>. Acesso em: 30 jun. 2016.

ANDRADE et al. REMD Mobile: repositório de materiais digitais móvel adaptado ao estilo de aprendizagem do aluno. **Revista Interdisciplinar de Ensino, Pesquisa e Extensão**, v. 1, n. 1, p. 68-80, 2013. Disponível em: <<http://revistaelectronica.unicruz.edu.br/index.php/electronica/article/download/68-80/pdf>>. Acesso em: 12 set. 2016.

ARAÚJO JÚNIOR. et al. Quão distantes estão as metodologias emergentes centradas na geração y, em relação aos nativos efetivamente digitais: reflexes e apontamento de soluções sob a ótica da estratégia de evolução adaptativa. In. V CONGRESSO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 2016. **Anais do XXII Workshop de Informática na Escola**, 2016. p. 973-977.

CORONASDK. **Corona documentation**. Disponível em: <<https://docs.coronalabs.com/>>. Acesso em: 11 jun. 2016.

COWAN, N. The magical number 4 in short-term memory: a reconsideration of mental storage capacity. **Behavioral and Brain Sciences**, v. 24, n. 1, p. 87-185, 2000. Disponível em: <<http://memory.psych.missouri.edu/doc/articles/2001/Cowan%20BBS%202001.pdf>>. Acesso em: 05 jun. 2016.

DIAS, D. C.; COSTA, N. S.; LIMA, L. V. Projeto e implementação de recursos didáticos multimídia interativos para melhoria do ensino de disciplinas de engenharia de computação. In: ICECE'2009 INTERNATIONAL CONFERENCE ON ENGINEERING AND COMPUTER EDUCATION, 2009, Buenos Aires. **Educating Engineers for Innovation**, Rio de Janeiro: COPEC, 2009, v. 1, p. 84-89.

FERREIRA, D. C. M. et al. Mapas de conhecimento estruturado: proposta de uma nova abordagem metodológica de ensino e aprendizagem. **Revista Educere et Educare**, Cascavel, v. 9, n. especial, p. 505-514, jul./dez. 2014.

HECKE, C. **Segundo Ibope, brasileiro passa 1h24 por dia usando smartphone**, abr. 2013. Disponível em: <<https://www.tecmundo.com.br/mercado/38863-segundo-ibope-brasileiro-passa-1h24-por-dia-usando-smartphone.htm>>. Acesso em: 25 ago. 2016.

INNSBRUCK, T. H. Micro learning and narration: exploring possibilities of utilization of narrations and storytelling for the designing of “micro units” and didactical micro-learning arrangements. **Research Gate**, p. 1-13, may. 2005. Disponível em: <<https://www.researchgate.net/publication/237558117>>. Acesso em: 12 fev. 2017.

JOHNSON, C. Thinking Differently About Mobile Learning. **Association for Talent Development**, USA, sept. 2008. Disponível em: <<https://www.td.org/Publications/Newsletters/Learning-Circuits/Learning-Circuits-Archives/2008/09/Thinking-Differently-About-Mobile-Learning>>. Acesso em: 21 set. 2016.

JOHNSON, C.; LOMAS, C. Design of the learning space: learning and design principles. **Educause Review**, Arizona, v. 40, n. 4, p. 16-28, july/aug. 2005. Disponível em: <<http://er.educause.edu/~media/files/article-downloads/erm0540.pdf>>. Acesso em: 26 jan. 2017.

KIRCOVE, B. **Android, iOS ou Windows Phone: qual é o melhor sistema para smartphones?** abr. 2013. Disponível em: <<http://www.techtudo.com.br/artigos/noticia/2013/04/android-ios-ou-windows-phone-qual-e-o-melhor-sistema-para-smartphones.html>>. Acesso em: 05 dez. 2016.

KOOLE, M. L. **A model for framing mobile learning**. Research Gate, p. 25-47, mar. 2009. Disponível em: <<https://www.researchgate.net/publication/252714629>>. Acesso em: 12 fev. 2017.

KUKULSKA-HULME, A.; SHARPLES, M. Mobile and contextual learning. **ALT-J: Research in Learning Technology**, v. 17, n. 3, p. 159-160, nov. 2009.

LEMOS, A. Dispositivos de leitura eletrônicos. **Comunicação, Mídia e Consumo**, São Paulo, v. 9, n. 24, p. 115-131, maio 2012. Disponível em: <<http://revistacmc.espm.br/index.php/revistacmc/article/view/239/234>>. Acesso em: 21 mar. 2017.

LEVY, P. **O que é o virtual?** São Paulo: Editora 34, 1996. 160 p.

MAYER, R. **Multimedia learning**. Cambridge: Cambridge University Press, 2001. 210 p.

MEDINA, J. **Brain rules: 12 principles for surviving and thriving at work, home, and school**. Pear Press, 2009. 320 p.

MEDINA, N. O.; FREITAS FILHO, P. J. Desenvolvimento do pensamento crítico na escrita colaborativa. **Revista Renote: Novas Tecnologias na Educação**, v. 2, n. 2, p. 1-11, 2004. Disponível em: <<http://www.seer.ufrgs.br/renote/article/viewFile/13789/15701>>. Acesso em: 06 set. 2016.

MESQUITA, I. C. A.; CONDE, M. G. A evolução gráfica do livro e o surgimento dos e-books. In: X CONGRESSO DE CIÊNCIAS DA COMUNICAÇÃO NA REGIÃO NORDESTE, 2008, São Luís. **Intercom – Sociedade Brasileira de Estudos Interdisciplinares da Comunicação**, 2008, p. 1-6

MILLER, G. The magical number seven, plus or minus two: some limits on our capacity for processing information. **Psychological Review**, v. 101, n. 2, p. 81-97, 1956. Disponível em: <<http://www.psych.utoronto.ca/users/peterson/psy430s2001/Miller%20GA%20Magical%20Seven%20Psych%20Review%201955.pdf>>. Acesso em: 03 mar. 2017.

NUNES, M. P.; GIRAFFA, L. M. M. **A Educação na Ecologia Digital**. PPGCC/FACIN, PUCRS, 2003.

OLIVEIRA, S. I. **Por que adotar mobile learning para treinamento?** out. 2016. Disponível em: <<http://www.proxxima.com.br/home/proxxima/how-to/2016/10/28/por-que-adotar-mobile-learning-para-treinamento.html>>. Acesso em: 25 nov. 2016.

OCKHAM, G. de. **Lógica dos Termos**. Porto Alegre: EDIPUCRS, 1999.

PAAS, F.; RENKL, A.; SWELLER, J. **Cognitive load theory: a special issue of educational psychologist**. Lawrence Erlbaum Associates, Incorporated, 2003. 76 p.

PHONEGAP. **Create your app with PhoneGap**. Disponível em: <<https://phonegap.com/>>. Acesso em: 11 jun. 2016.

RAZ, A.; FAN, J.; POSNER, M. I. Hypnotic suggestion reduces conflict in the human brain. **Proceedings of the National Academy of Sciences**, United States of America, v. 102, n. 28, p. 9978-9983, 2005.

RUGGIERO, W. V. et al. TIDIA Ae: An Electronic Learning System. In: **3RD INTERNATIONAL SYMPOSIUM IN DIGITAL LIBRARY**, 2005, São Paulo. **Proceedings of the 3rd International Symposium in Digital Library**, São Paulo: ISTEC/IEEE, v. 1, 2005.

SAMBATECH. **Materiais Gratuitos: Bee Smart**. Disponível em: <<http://sambatech.com/bee-smart/>>. Acesso em: 05 dez. 2016.

SANTOS, L. M. A.; TAROUCO, L. M. R. A importância do estudo da teoria da carga cognitiva em uma educação tecnológica. **Revista Renote: Novas Tecnologias na Educação**, v. 5, n. 1, p. 1-9, jul. 2007. Disponível em: <<http://www.seer.ufrgs.br/renote/article/download/14145/8082>>. Acesso em: 15 fev. 2017.

STRAYER, D. L. et al. Measuring Cognitive Distraction in the Automobile II: Assessing In-Vehicle Voice-Based Interactive Technologies. **AAA Foundation for Traffic Safety**, Washington, p. 1-52, 2013.

TELECO: INTELIGÊNCIA EM TELECOMUNICAÇÕES. **Estatísticas de celulares no Brasil**. Disponível em: <<http://www.teleco.com.br/ncl.asp>>. Acesso em: 08 abr. 2016.

WAYCOTT, J.; HULME, A. Student's experiences with PDAs for reading course materials. **Personal and Ubiquitous Computing**, v. 7, n. 1, p. 30-43, 2003.

XAMARIN. **Xamarin: Deliver native Android, iOS, and Windows apps, using existing skills, teams, and code.** Disponível em: <<https://www.xamarin.com/>>. Acesso em: 11 jun. 2016.

XAVIER, A. C. Letramento digital: impactos das tecnologias na aprendizagem da Geração Y. **Calidoscópio**, v. 9, n. 1, p. 3-14, jan./abr. 2011. Disponível em:<<http://revistas.unisinos.br/index.php/calidoscopio/article/view/748/149>>. Acesso em: 12 dez. 2016.