

Informática Aplicada à Educação

Everson Mario Novak
(Organizador)



 **Editora**
Atena

Ano 2018

Everson Mario Novak
(Organizador)

Informática Aplicada à Educação

Atena Editora
2018

2018 by Atena Editora

Copyright © da Atena Editora

Editora Chefe: Profª Drª Antonella Carvalho de Oliveira

Edição de Arte e Capa: Geraldo Alves e Natalia Sandrini

Revisão: Os autores

Conselho Editorial

Prof. Dr. Alan Mario Zuffo – Universidade Estadual de Mato Grosso do Sul
Prof. Dr. Álvaro Augusto de Borba Barreto – Universidade Federal de Pelotas
Prof. Dr. Antonio Carlos Frasson – Universidade Tecnológica Federal do Paraná
Prof. Dr. Antonio Isidro-Filho – Universidade de Brasília
Prof. Dr. Constantino Ribeiro de Oliveira Junior – Universidade Estadual de Ponta Grossa
Profª Drª Daiane Garabeli Trojan – Universidade Norte do Paraná
Profª Drª Deusilene Souza Vieira Dall’Acqua – Universidade Federal de Rondônia
Prof. Dr. Eloi Rufato Junior – Universidade Tecnológica Federal do Paraná
Prof. Dr. Fábio Steiner – Universidade Estadual de Mato Grosso do Sul
Prof. Dr. Gianfábio Pimentel Franco – Universidade Federal de Santa Maria
Prof. Dr. Gilmei Fleck – Universidade Estadual do Oeste do Paraná
Profª Drª Girlene Santos de Souza – Universidade Federal do Recôncavo da Bahia
Profª Drª Ivone Goulart Lopes – Istituto Internazionele delle Figlie de Maria Ausiliatrice
Prof. Dr. Jorge González Aguilera – Universidade Federal de Mato Grosso do Sul
Prof. Dr. Julio Candido de Meirelles Junior – Universidade Federal Fluminense
Profª Drª Lina Maria Gonçalves – Universidade Federal do Tocantins
Profª Drª Natiéli Piovesan – Instituto Federal do Rio Grande do Norte
Profª Drª Paola Andressa Scortegagna – Universidade Estadual de Ponta Grossa
Profª Drª Raissa Rachel Salustriano da Silva Matos – Universidade Federal do Maranhão
Prof. Dr. Ronilson Freitas de Souza – Universidade do Estado do Pará
Prof. Dr. Takeshy Tachizawa – Faculdade de Campo Limpo Paulista
Prof. Dr. Urandi João Rodrigues Junior – Universidade Federal do Oeste do Pará
Prof. Dr. Valdemar Antonio Paffaro Junior – Universidade Federal de Alfenas
Profª Drª Vanessa Bordin Viera – Universidade Federal de Campina Grande
Prof. Dr. Willian Douglas Guilherme – Universidade Federal do Tocantins

Dados Internacionais de Catalogação na Publicação (CIP) (eDOC BRASIL, Belo Horizonte/MG)	
143	Informática aplicada à educação [recurso eletrônico] / Organizador Everson Mario Novak. – Ponta Grossa (PR): Atena Editora, 2018. 10.596 kbytes Formato: PDF Requisitos de sistema: Adobe Acrobat Reader Modo de acesso: World Wide Web Inclui bibliografia ISBN 978-85-85107-14-7 DOI 10.22533/at.ed.147181308 1. Educação. 2. Informática. 3. Tecnologia educacional. I. Novak, Everson Mario. CDD 371.334
Elaborado por Maurício Amormino Júnior – CRB6/2422	

O conteúdo do livro e seus dados em sua forma, correção e confiabilidade são de responsabilidade exclusiva dos autores

2018

Permitido o download da obra e o compartilhamento desde que sejam atribuídos créditos aos autores, mas sem a possibilidade de alterá-la de nenhuma forma ou utilizá-la para fins comerciais.

www.atenaeditora.com.br

E-mail: contato@atenaeditora.com.br

APRESENTAÇÃO

Este livro foi dividido em 3 eixos, fruto de pesquisa científica de ótima qualidade acadêmica sobretudo por equipes multidisciplinares e de diversas instituições. Os trabalhos realizados são para auxiliar na Educação a distância e presencial, utilizando recursos computacionais para o planejamento e desenvolvimento de aplicativos para apoiar o aprendizado de matemática e de atividades cotidianas para crianças autistas, desenvolvimento de jogos educacionais e ainda para avaliar os dados armazenados em LMS (Learning Management Software) da plataforma Moodle.

No primeiro eixo temos o desenvolvimento de softwares e aplicativos voltados para a EAD, iniciamos por uma aplicação m-learning Genius para o auxiliar no ensino de matemática na educação infantil, explorando formas geométricas, números e a adição e subtração através de figuras e sons. O ENEN foi tema de um aplicativo focado em preparar os alunos na disciplina de matemática. O relacionamento social, comunicação e alterações de comportamento do autista são o tema de estudo para o desenvolvimento de um aplicativo para auxiliar os autistas no aprendizado e no relacionamento social.

A Cloud Computing apoia a aprendizagem em ambientes U-learning para verificar os estilos de aprendizagem e aplicabilidade em ambientes educacionais. As métricas de software são utilizadas para fazer uma análise da aprendizagem em cursos de programação a distância. Uma base de conhecimento gerada das questões e códigos inseridos nas plataformas digitais de ensino, foi feita a classificação de códigos da linguagem C em medidas similares para fazer os agrupamentos para formação de uma base de questões com códigos e soluções associadas para correções de questões de forma automatizada.

O segundo eixo entra em jogos digitais e gamificação, auxiliam na aprendizagem de pessoas com deficiência visual, tenta garantir no processo pedagógico uma inclusão digital e social destas pessoas. O processo de aprendizado utilizou-se dos jogos construcionistas para propor quatro jogos educativos, simplificando a complexidade na sua criação. Problemas motivacionais dos alunos são tratados na gamificação para verificar o que ocorre em processos de aprendizagem em ambientes educacionais.

No terceiro e último eixo é abordada a aprendizagem de máquina (machine-learning), aplicada a educação e aprendizado. O conceito de Estilos de Aprendizagem (EA) da psicologia cognitiva e da pedagogia, são propostos em sistemas educacionais adaptativos, com algumas aplicações da Aprendizagem por Reforço, foi proposto uso de algoritmos relacionados a aprendizagem de máquina para obter os estilos de Aprendizagem. Aplicabilidade de modelos de Regressão Múltipla no contexto da EAD foi abordado para validar as variáveis de comportamento de autorregulação da aprendizagem na plataforma LMS – Moodle.

Ao escrever este prefácio contextualizei o alinhamento das análises e teorias desenvolvidas nos artigos contidos neste livro. Sugiro que o leitor faça este caminho para uma compreensão ampla destes trabalhos, agradeço a oportunidade de fazer parte de grupo e felicito a todos os integrantes.

Everson Mario Novak
Mestrando em Informática - PUCPR

SUMÁRIO

EIXO 1: SOFTWARES E APLICATIVOS VOLTADOS PARA A EAD

CAPÍTULO 1	1
GENIUS MATH: UMA APLICAÇÃO MOBILE PARA AUXILIAR A APRENDIZAGEM DA MATEMÁTICA NA PRÉ-ESCOLA	
<i>Stefane Vieira Menezes</i> <i>Jiani Cardoso da Roza</i>	
CAPÍTULO 2	13
APLICATIVO MÓVEL PARA PREPARAÇÃO DE ESTUDANTES PARA O ENEM NO CONTEXTO DA DISCIPLINA DE MATEMÁTICA	
<i>Hannderson Faria Arantes</i> <i>Rodrigo Duarte Seabra</i>	
CAPÍTULO 3	27
COTIDIANO: UM SOFTWARE PARA AUXILIAR CRIANÇAS AUTISTAS EM SUAS ATIVIDADES DIÁRIAS	
<i>Afranio Furtado de Oliveira Neto</i> <i>Hugo Leonardo Pereira Rufino</i> <i>Diovane de Godoi Beira</i> <i>Rodolfo Bocado Palis</i> <i>Paula Teixeira Nakamoto</i>	
CAPÍTULO 4	41
APRENDIZAGEM SIMULADA NA NUVEM	
<i>Rafaela R. Jardim</i> <i>Roseclea Duarte Medina</i> <i>Giliane Bernardi</i> <i>Fabricio Herpich</i> <i>Andressa Facalde</i> <i>Eduardo Lemos</i>	
CAPÍTULO 5	55
ANÁLISE DA APRENDIZAGEM DE PROGRAMAÇÃO POR MAPEAMENTO DE PERFIS EM MÉTRICAS DE SOFTWARE	
<i>Márcia Gonçalves de Oliveira</i> <i>Ádler Oliveira Silva Neves</i> <i>Helen França Medeiros</i> <i>Mônica Ferreira Silva Lopes</i> <i>Leonardo Leal Reblin</i> <i>Elias Silva de Oliveira</i>	
CAPÍTULO 6	68
CLASSIFICAÇÃO DE CÓDIGOS C USANDO MEDIDAS DE SIMILARIDADE PARA APOIO AO ENSINO DE PROGRAMAÇÃO	
<i>José Carlos Campana Filho</i> <i>Elias Silva de Oliveira</i> <i>Márcia Gonçalves de Oliveira</i>	

EIXO 2: JOGOS DIGITAIS E GAMIFICAÇÃO

CAPÍTULO 7 79

BEM EXPRESSÕES: JOGO DIGITAL VOLTADO PARA O ENSINO INCLUSIVO DA MATEMÁTICA

André Luis Bitencourt Fernandes
Claudia Pinto Pereira
Kayo Costa de Santana
Ana Jaize de Oliveira Silva Santos
Bruno Gonzaga de Mattos Vogel

CAPÍTULO 8 95

JINDIE: UMA LINHA DE PRODUTO DE SOFTWARE PARA JOGOS EDUCATIVOS COM FOCO NO CONSTRUCIONISMO

Carlos Alberto Correia Lessa Filho
Arturo Hernandez Dominguez

CAPÍTULO 9 107

METODOLOGIAS GAMIFICADAS PARA A EDUCAÇÃO: UMA REVISÃO SISTEMÁTICA
DETECÇÃO AUTOMÁTICA DE ESTILOS DE APRENDIZAGEM: UMA ANÁLISE COMPARATIVA DE
CLASSIFICADORES APLICADOS EM UM CENÁRIO REAL DE APRENDIZADO

André Luiz de Souza Brito
Charles Andryê Galvão Madeira

EIXO 3: APRENDIZAGEM DE MÁQUINA APLICADA A EDUCAÇÃO

CAPÍTULO 10 120

DETECÇÃO AUTOMÁTICA DE ESTILOS DE APRENDIZAGEM: UMA ANÁLISE COMPARATIVA DE
CLASSIFICADORES APLICADOS EM UM CENÁRIO REAL DE APRENDIZADO

Lucas Daniel Ferreira
José Fernando Rodrigues Jr

CAPÍTULO 11 140

DETECÇÃO AUTOMÁTICA E DINÂMICA DE ESTILOS DE APRENDIZAGEM EM SISTEMAS
ADAPTATIVOS E INTELIGENTES PARA A EDUCAÇÃO UTILIZANDO DYNAMIC SCRIPTING

Júlio César da Costa Silva
Cristiano Grijó Pitangui
Alessandro Vivas Andrade
Luciana Pereira de Assis
Cristiano Maciel da Silva

CAPÍTULO 12 156

UM PROCESSO DE VALIDAÇÃO DE VARIÁVEIS COMPORTAMENTAIS DE AUTORREGULAÇÃO
DA APRENDIZAGEM EM PLATAFORMAS DE LMS

Rodrigo Lins Rodrigues
João Carlos Sedraz Silva
Jorge Luis Cavalcanti Ramos
Fernando da Fonseca de Souza
Alex Sandro Gomes

SOBRE O ORGANIZADOR..... 166

EIXO 1 – SOFTWARES E APLICATIVOS VOLTADOS PARA A EAD

APRESENTAÇÃO

No primeiro eixo temos o desenvolvimento de softwares e aplicativos voltado para EAD, iniciamos por uma aplicação m-learning Genius para o auxiliar no ensino de matemática na educação infantil, explorando formas geométricas, números e a adição e subtração através de figuras e sons. Com atividades lúdicas viabilizando práticas contemporâneas ao cotidiano infantil.

Agora abordando outro tema pertinente o ENEN, um aplicativo focado em preparar os alunos para o Exame Nacional do Ensino Médio na disciplina de matemática.

As dificuldades apresentadas em relacionamento social, comunicação e alterações de comportamento por um autista são o tema de estudo para o desenvolvimento de um aplicativo para auxiliar os autistas no aprendizado e no relacionamento social.

A Cloud Computing está apoiando a aprendizagem em ambientes U-learning, criando um laboratório virtual U-Lab Cloud para verificar os estilos de aprendizagem para adotar a tecnologia em ambientes educacionais.

O software PCódigo II, utiliza métricas de software para fazer a análise da aprendizagem em cursos de programação a distância, para que sejam observadas dificuldades de aprendizagem, boas práticas de programação e perfis de aprendizagem de forma rápida, detalhada e holística.

Neste outro tema é gerado uma base de conhecimento de forma organizada das questões e códigos gerados nas plataformas digitais de ensino a distância. Abordando uma classificação de códigos da linguagem C baseada em medidas similares para fazer os agrupamentos para formação de uma base de questões com códigos e soluções associadas para correções de questões de forma automatizada.

Everson Mario Novak
Mestrando em Informática - PUCPR

GENIUS MATH: UMA APLICAÇÃO MOBILE PARA AUXILIAR A APRENDIZAGEM DA MATEMÁTICA NA PRÉ-ESCOLA

Stefane Vieira Menezes

Instituto Federal de Educação, Ciência e
Tecnologia Farroupilha
Alegrete – RS

Jiani Cardoso da Roza

Instituto Federal de Educação, Ciência e
Tecnologia Farroupilha
Alegrete – RS

RESUMO Este artigo visa apresentar uma aplicação de *m-learning* denominado *Genius Math* voltado a apoiar o ensino e a aprendizagem da matemática na educação infantil. Ao considerarmos que na educação infantil muitas aprendizagens se dão a partir da curiosidade e do entusiasmo das crianças, o aplicativo desenvolvido na linguagem de programação Java e voltado para celulares com sistema operacional Android busca explorar o conhecimento dos números, formas geométricas e a iniciação as operações de adição e subtração apoiado em figuras e sons. Desta forma, através de atividades lúdicas para as crianças e de atividades planejadas pelo professor com o uso das tecnologias digitais pode-se viabilizar práticas pedagógicas contemporâneas ao cotidiano infantil. Para testar o aplicativo foi realizado um ensaio de interação em sala de aula por crianças da educação infantil que, em grupos formados aleatoriamente, brincaram com as diferentes atividades matemáticas propostas

no aplicativo.

PALAVRAS-CHAVE jogos educacionais, matemática na pré-escola, aplicativo *mobile*.

ABSTRACT This article presents an application of m-learning called *Genius Math* aimed to support the teaching and learning of mathematics in early childhood education. When we consider that in early childhood education learning many are given from the curiosity and enthusiasm of children, the application developed in the Java programming language and geared Mobile Android operating system aims to explore the knowledge of numbers geometric shapes and initiating the operations of addition and subtraction supported by pictures and sounds. In this way, through playing activities for children and activities planned by the teacher with the use of digital technologies can enable contemporary pedagogical practices to children every day. To test the application was made an interaction test in the classroom for children of early childhood education that in randomly formed groups, they played with different mathematical activities proposed in the application.

1 | INTRODUÇÃO

As Tecnologias de Informação e Comunicação (TIC) tem oportunizado às pessoas formas cada vez mais dinâmicas e atrativas de comunicação e interação social. Aos poucos estamos abandonando nossa condição passiva frente as diversas atividades e serviços cotidianos realizados, com facilidade em um curto espaço de tempo e lugar através de uma tecnologia digital tal como o: celular, *tablet* ou computador.

Segundo uma pesquisa realizada pelo IBGE (2010) sobre o uso de tecnologias da informação pelos brasileiros, o acesso à internet pelo celular ultrapassou o acesso à internet pelo computador. Neste sentido, reconhecendo o potencial dos dispositivos móveis na atualidade para interação social, serviços e entretenimento e, sendo o celular uma mídia com capacidade de penetração social ampla, basicamente amparada na comunicação oral e de custo acessível, torna-se oportuno explorar cada vez mais a utilização do celular no âmbito educacional.

O conjunto de atividades a serem exploradas no *Genius Math* toma por base as atividades matemáticas apresentadas nos materiais do projeto Buriti Mirim 2 para a Educação Infantil da editora Moderna. O projeto Buriti Mirim é adotado em algumas instituições nacionais com oferta de pré-escola, sendo assim, ao identificarmos que uma instituição da cidade de Alegrete também o utilizava como material de apoio pedagógico e considerando que o mesmo contempla em seus diferentes eixos de aprendizagem a matemática optamos por utilizá-lo como estratégia de ensino.

A atividade lúdica é fundamental para subsidiar o processo de construção do conhecimento e por isso é indispensável na prática educativa (SANTOS, 2011). Nesta intenção, as atividades desenvolvidas no aplicativo são realizadas considerando a necessidade das crianças de aprenderem brincando, neste caso brincando no celular, com figuras e formas geométricas.

Genius Math está desenvolvido em Java e é um aplicativo para celulares com sistema operacional Android. O desenvolvimento do aplicativo foi acompanhado por duas professoras da educação infantil que enfatizaram a necessidade de brincar e explorar os números cardinais e as operações aritméticas de adição e subtração. Após o desenvolvimento, o aplicativo foi submetido a uma experimentação através de testes de interação com o usuário, neste caso crianças da educação infantil de 5 a 6 anos.

2 | O GENIUS MATH NA PERSPECTIVA DE UM APLICATIVO EDUCACIONAL

Pesquisas realizadas mostram que há um constante crescimento do uso da TIC no Brasil no âmbito educacional (DE SOUZA MONTEIRO *et al.*, 2015). Da mesma forma, é cada vez mais comum o acesso a aplicativos educacionais disponibilizados através de dispositivos móveis, conhecidos vulgarmente por *Apps*, um forte representante das TIC atuais. A interação através de dispositivos móveis traz formas de interação

mais dinâmicas e capazes de expandir as relações professor-estudante e estudante-estudante dentro e fora das escolas, uma vez que a interação pode alterar as relações tanto no tempo (síncrona e assíncrona) (HRASTINSKI, 2008) quanto no espaço (presencial e a distância) (NORTVIG, 2014).

Neste sentido, o aprendizado móvel (ou simplesmente *m-learning*) pode ser entendido como uma parte específica do aprendizado eletrônico (*e-learning*) para dispositivos móveis (GEORGIEV et al., 2004). O *m-learning* se popularizou a partir do barateamento e acesso facilitado aos dispositivos móveis, dentre eles os celulares que hoje disponibilizam diversos recursos integrados tais como câmera, GPS, tela sensível ao toque e principalmente acesso à Internet, facilitando a obtenção de aplicativos para diferentes finalidades por qualquer usuário.

Desta forma, o *Genius Math* é uma das diversas iniciativas de utilizar a tecnologia digital, já presente no dia a dia das crianças, com um fim educativo e de entretenimento, pois a criança aprende brincando e interagindo com as outras.

3 | TRABALHOS RELACIONADOS

Visando coletar informações de *layout*, conteúdo e abrangência, entre outras características a serem consideradas no *Genius Math*, foi conduzida uma busca sistemática na *PlayStore* do *Google*, por dois motivos: primeiramente por ser uma loja virtual que disponibiliza o *download* gratuito a *Apps* para um acervo de aplicativos que podem ser consultados/utilizados por qualquer usuário de celular, inclusive crianças, que com poucos cliques realizam o *download*. A busca foi refinada em diversas formas: (1) Categoria – *Apps* educativos, Subcategorias – *Apps* gratuitos, matemática para crianças, em todas as classificações (número de estrelas de classificação que o aplicativo recebe dos usuários); (2) Categoria – Educativos, Subcategorias – em todas as classificações; (3) Categoria – Jogos e *Apps* populares, para crianças até 5 anos de idade (uma vez que a outra categoria disponível oferece para crianças de 6 a 8 anos de idade).

Nas três buscas realizadas com diferentes refinamentos foi possível perceber que os *Apps* resultantes da primeira categoria estavam mais próximos do que se pesquisava, uma vez que nas demais categorias já pré-configuradas na *PlayStore* retornavam aplicativos para diferentes aprendizagens em física, português, inglês, entre outros. Assim, no primeiro grupo retornaram mais de oitenta aplicativos, porém muitos deles não estabeleciam a faixa etária atendida apenas a finalidade no contexto de ensino e aprendizagem da matemática. Desta forma, foi necessário clicar na descrição desses aplicativos para avaliar os que estivessem próximos a proposta do *Genius Math* e que na descrição se dizia voltado ao aprendizado da matemática na pré-escola ou educação infantil. Assim, foram analisados os seguintes aplicativos:

Soma para crianças: Seu objetivo é a realização de cálculos de adição. Suas

opções de jogos são controladas por tempo, sendo possível a escolha entre um ou dois minutos para a realização dos cálculos, o usuário deve realizar o maior número possível de cálculos no tempo determinado, quando este acaba são informados quantos resultados positivos e negativos o usuário fez. Em um teste rápido foi possível perceber que o maior resultado possível seria oito, pois os cálculos são relacionados às imagens e estas não suportam mais do que quatro elementos em cada, além do fato de todos os *feedbacks* serem em inglês.

Educational Math for Kids: São apresentados cálculos de adição e subtração para o usuário e este deve selecionar o resultado correto, não existe uma quantidade máxima de cálculos a serem resolvidos, porém o usuário tem três chances de acertar o resultado correto, caso não o faça, é informado que o jogo acabou e o jogo é iniciado novamente. Há a possibilidade de aumentar a dificuldade através das três modalidades em que o aplicativo divide-se. Verificou-se que não é informado para o usuário previamente que o jogo tem um controle de tentativa e erro, onde o usuário tem três chances de erro antes do jogo ser reiniciado, também pode-se citar o fato de que não há um número máximo de cálculos à serem realizados e por este motivo, muitas vezes os cálculos se repetem, além de que o sistema é em inglês e o seu nível de dificuldade requer que o usuário já tenha certo conhecimento da matemática, pois em todos os seus níveis são apresentados cálculos que resultam em números negativos. Na primeira utilização do aplicativo foi sugerida a migração para a sua versão paga e foi avisado que esta versão estava obsoleta, ao fim deste estudo a versão grátis já não estava mais disponível na *PlayStore*.

Matemática para crianças Soma: conhecido como Matemática das maçãs, em sua descrição na *PlayStore* menciona seu benefício às crianças em fase de aprendizado inicial da matemática. Tem como foco a operação de adição e como objetivo a resolução de cálculos simples que resultem no máximo em dez. São apresentados para as crianças valores e o resultado de zero até dez, todos os números têm representações de maçãs em seus respectivos valores, tem total interação de voz em português. Um fato observado esta aplicação é a demora entre os cálculos e durante sua execução, o que pode fazer com que as crianças percam o interesse ao ter que esperar.

Nos aplicativos analisados foi notada a falta de níveis diferentes de aprendizagem nas atividades disponíveis. Desta forma, o *Genius Math* busca incluir em suas atividades diferentes níveis de dificuldade. A tabela 1 mostra um comparativo dos aplicativos analisador e do aplicativo *Genius Math*.

	Idioma Português	Diferentes níveis de dificuldade	Direcionado a crianças da pré-escola	Música	Letra maiúscula	Gratuito
Soma para crianças			√			√
Educational math for kids		√			√	

Matemática para crianças soma	✓		✓			✓
Genius math	✓	✓	✓	✓	✓	✓

Tabela 1: Comparação entre os softwares disponíveis e o Genius Math

FONTE: Adaptado de TAJRA (2001).

4 | O APLICATIVO GENIUS MATH

O aplicativo *Genius Math* tem como intuito proporcionar a possibilidade do conhecimento junto ao entretenimento. Com foco na matemática da pré-escola, o aplicativo disponibiliza dois módulos distintos, sendo um deles com foco na quantificação e o outro em operações aritméticas de adição e subtração, usando ilustrações para facilitar o entendimento das atividades.

4.1 Metodologia utilizada

A proposta de implementar um aplicativo com fins educacionais para apoiar a aprendizagem da matemática surgiu a partir de uma pesquisa exploratória que segundo GIL (2002) visa proporcionar maior familiaridade com o problema. Desta forma, uma pesquisa exploratória foi realizada visando identificar o uso das tecnologias digitais nas séries iniciais e educação infantil em algumas escolas da cidade. Percebeu-se que boa parte das escolas levam as crianças até os laboratórios de informática, semanalmente ou quinzenalmente para acessar sítios infantis com diferentes finalidades de aprendizagem. Porém, o uso de aplicativo em celular como instrumento de aprendizagem não foi identificado nas escolas visitadas.

Com estes dados, partiu-se para a busca de materiais para conceder o apoio pedagógico necessário para a criação de um protótipo adequado para a educação infantil. Percebeu-se com as visitas realizadas que algumas escolas adotavam como material pedagógico de apoio os livros do projeto Buriti Mirim (EDITORA MODERNA, 2015). Decidimos então, utilizar tais livros como fonte pedagógica para criação das imagens (figuras de animais, formas geométricas, números, frutas) e das atividades, com o diferencial que a interação com o aplicativo se dá através do toque, do colorir passando o dedo sobre uma imagem, do apagar ao tocar no cancelar, etc. Nas diferentes atividades disponibilizadas no *Genius Math* foi utilizado o mínimo de texto possível pois boa parte das crianças de quatro a seis anos ainda não são totalmente alfabetizadas. Também por esse motivo, acreditamos que o *Genius Math* é um aplicativo educacional para ser utilizado como o apoio do professor ou dos pais, pelo menos até o reconhecimento do aplicativo por parte das crianças.

A aplicação foi desenvolvida para a plataforma *Android* por vários fatores: a) familiaridade das crianças com o uso de celulares; b) possibilidade de utilizada do *App off-line* após o download; c) o desejo da desenvolvedora de trabalhar com aplicações

móveis para *Android*. Neste sentido foram utilizadas as seguintes ferramentas: *Android Studio*, *Java*, *BALSAMIQ* e *Astah UML*.

O desenvolvimento do aplicativo seguiu o processo interativo de geração de modelos a partir da abordagem de prototipação evolutiva conforme definições de Presmann (2011). Quanto a arquitetura e modelagem do sistema o mesmo foi desenvolvido a partir de dois casos de usos principais guiando-se pelos conceitos de Fowler (2014) para modelagem de sistemas orientados a objetos. Desta forma, temos: 1) “contar figuras” e 2) “calcular figuras”. O “contar figuras” pode ser estendido pelos casos de usos Selecionar número e pintar figura. Já o caso de uso “calcular figura” inclui o caso de uso “calcular número”. Um caso de uso “mostrar recompensa” pode expandir qualquer caso de uso.

Cada atividade (caso de uso) possuiu uma classe Java para a sua configuração que determina, através de métodos, quais serão as imagens e os valores utilizados. Para a realização da recompensa visual foi necessário a inserção de um arquivo *xml* com todas as imagens necessárias para a animação. A recompensa visual está presente ao final de cada atividade concluída com sucesso, para que isto acontecesse foi necessária a codificação de uma classe Java denominada Controlador.

4.2 Desenvolvimento

Na Figura 1 é apresentado a tela inicial da aplicação *Genius Math*, no detalhe 1, há um botão que controla a música tocada na aplicação, esta opção pode ser desabilitada em qualquer momento da sua utilização, pois é apresentada em todas as telas.

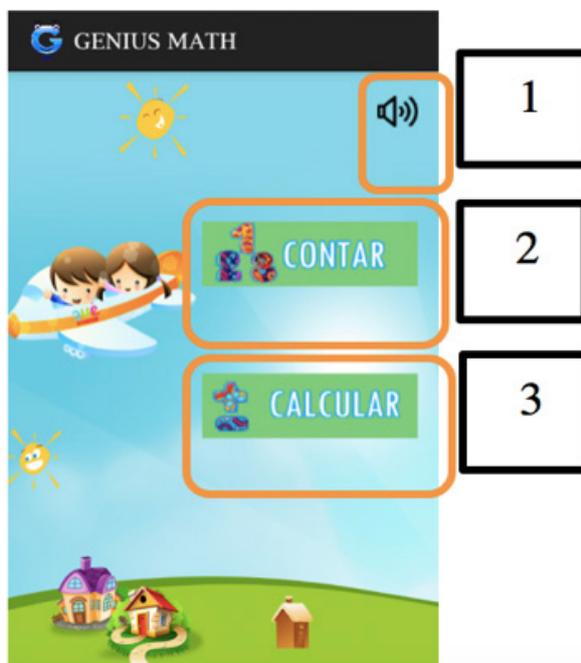


Figura 1. Tela inicial da aplicação Genius Math.

Na Figura 2, à direita, é apresentado o resultado do clique no botão Contar e à esquerda é apresentado o resultado do clique no botão Calcular, apresentados

nos detalhes 2 e 3 da Figura 1. Há três atividades disponíveis no módulo Contar, denominados: Contar animais; Contar Colorindo; Contar Formas Geométricas. E três atividades disponíveis no módulo Calcular, denominados: Adição; Subtração; Calculando com formas geométricas.

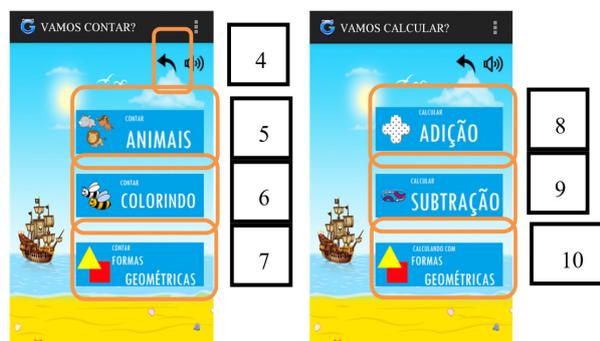


Figura 2. Tela resultado do clique no botão Contar ou Calcular.

No detalhe 4 é ilustrado o botão voltar, apresentado em todas as demais telas, proporcionando a possibilidade de voltar para a tela anterior, quando solicitado. O detalhe 5, Figura 2, quando clicado, destaca a atividade Contar Animais referente a Figura 3, à esquerda, essa atividade consiste na apresentação de uma imagem com determinados números de animais e três opções de respostas, sendo destas somente uma correta, ao clicar em um botão com o resultado errado o mesmo desaparece, uma mensagem é apresentada para o usuário e um som associado ao erro é emitido. Quando o resultado correto é selecionado é apresentada uma nova imagem, novas opções de resultados e é emitido um som para ser associado ao acerto.

Ao clicarmos no detalhe 6 da Figura 2, denominado Contar Colorindo, é apresentado para o usuário a tela ilustrada na Figura 3, ao centro, nesta atividade são apresentadas nove imagens de animais em preto e branco e a instrução de quantos animais deverão ser pintados, ao clicar em cima de qualquer uma das imagens a mesma fica colorida e se selecionada novamente volta ao estado inicial. Se for informado um resultado diferente do solicitado e o botão Continuar, em destaque, for clicado é apresentada uma mensagem para o usuário e não ocorre transição de telas até que o resultado correto seja selecionado.

Quando clicado no detalhe 7 da Figura 2, a atividade Contar Formas Geométricas é solicitada, ilustrada na Figura 3, à direita. Esta atividade consiste na apresentação de uma imagem com determinadas quantidades de triângulos, quadrados e círculos, três opções de resultados e a imagem de qual forma geométrica está sendo solicitada. O reconhecimento das formas geométricas faz parte da metodologia ensinada na matemática e é explorada pelo aplicativo.

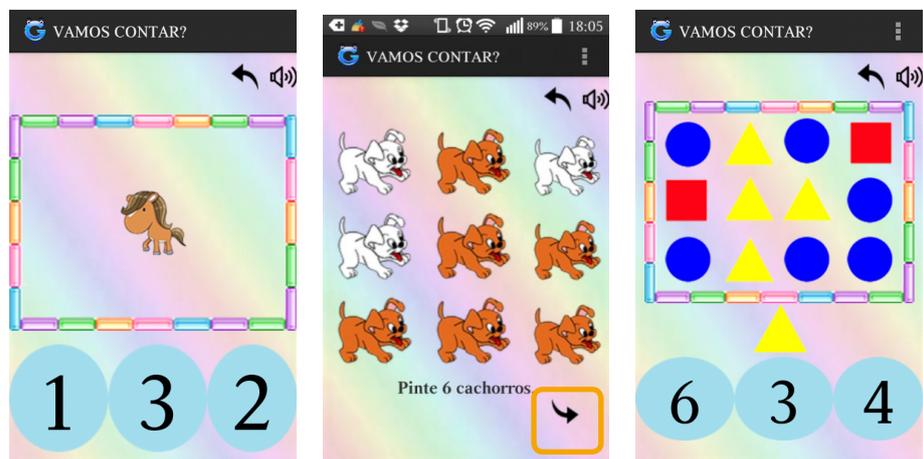


Figura 3. Atividade do módulo Contar

O detalhe 8, apresentado na Figura 2 à direita, se selecionado ilustra a atividade Adição, onde são apresentados pequenos cálculos exclusivamente desta operação, com resultados inferiores ao número dez. A Figura 5, à direita, exemplifica esta atividade, mostrando a utilização de animais para ilustrar os cálculos, além disso os cálculos são montados com algoritmos para melhor entendimento do usuário. São disponibilizados três resultados, sendo destes somente um correto, quando um resultado errado for selecionado o botão torna-se invisível, é emitido o som referente ao erro e uma mensagem é apresentada para o usuário. Ao selecionar o resultado correto é apresentado outros cálculos com novos resultados e um som é emitido para ser associado ao acerto.

O detalhe 9 apresentado na Figura 2, destaca o botão que solicita a atividade exclusiva da operação subtração, quando clicado. Esta atividade contém a mesma metodologia utilizada na atividade de Adição, e é ilustrada na Figura 4, ao centro. Nesta atividade nenhum dos cálculos solicitados tem valor maior que seis, pelo fato de que o público alvo deste aplicativo são crianças que estão na pré-escola e ainda não possuem um domínio claro da operação de subtração.

No detalhe 10 da Figura 2, podemos observar a atividade intitulada Calculando com Formas Geométricas, nesta atividade as operações matemáticas de adição e subtração são trabalhadas juntamente. O intuito desta metodologia é a distinção entre as formas geométricas e o raciocínio lógico necessário para realizar cálculos a partir destas formas. Como observamos na Figura 4, à direita, seu *layout* e suas funcionalidades são parecidos com as demais atividades explicadas anteriormente.

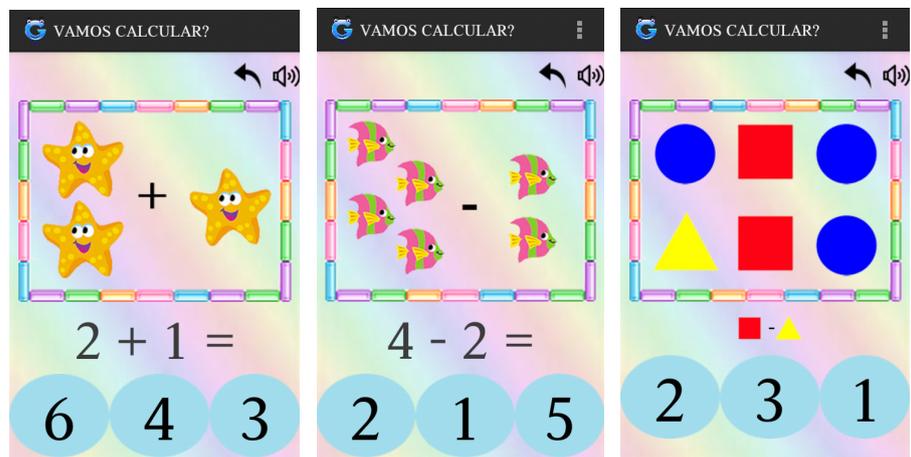


Figura 4. Atividades do módulo Calcular.

A Figura 5 traz as telas de recompensa visual e auditiva apresentada para o usuário a cada ciclo concluído de qualquer uma das atividades apresentadas anteriormente. Consiste em uma animação acompanhada do som de palmas. A mensagem que lhe é apresentada é modificada dependendo da sua quantidade de erros durante a execução da atividade. Caso a criança conclua as atividades com cem por cento de acertos lhe é apresentado a mensagem “VOCÊ É UM GÊNIO”, se ele obtiver um número inferior a três erros lhe é apresentado a mensagem “PARABÉNS”, se a margem de erro da criança estiver entre quatro e sete erros a mensagem é “LEGAL”, estes exemplos podem ser observados na Figura 5. As atividades têm em média dez exercícios cada, por este motivo foi escolhido estes parâmetros para serem utilizados nas mensagens, porém em atividades que tenham menos exercícios estes parâmetros foram adaptados.

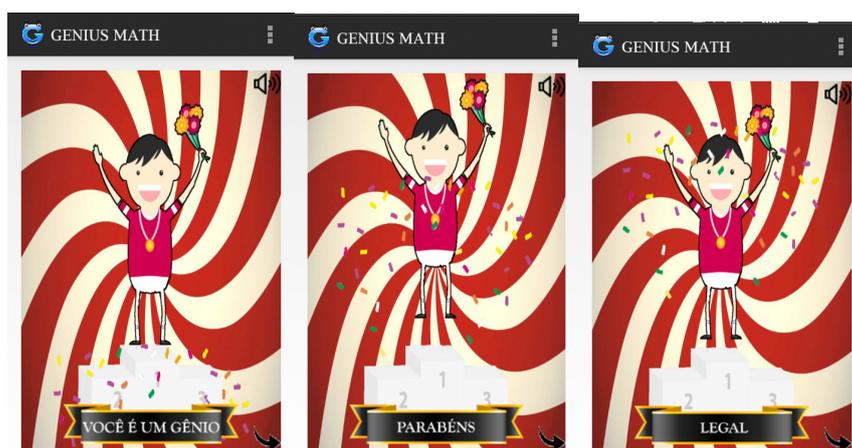


Figura 5. Tela recompensa apresentada ao final das atividades.

5 | AVALIAÇÃO EXPERIMENTAL

Com a primeira versão do aplicativo funcional foi realizada a primeira avaliação com uma professora da educação infantil a partir do uso da ferramenta diretamente em um celular. A partir das sugestões da professora, foram realizadas modificações, como

por exemplo, o tamanho da fonte e a preferência da utilização de letra maiúscula nos enunciados das atividades e o grau de dificuldade das atividades, onde os cálculos de adição deveriam ter maior grau de dificuldade do que os cálculos de subtração.

Após as modificações concluídas foi realizada uma segunda avaliação que consistiu em um ensaio de interação (NIELSEN, 1999) onde o aplicativo foi apresentado para o seu público alvo, com o intuito de observar a viabilidade do aplicativo em um ambiente real e detectar possíveis falhas ou ajustes a serem realizados. O aplicativo foi apresentado para vinte estudantes do terceiro ano da pré-escola de uma escola particular situada no centro da cidade de Alegrete. As crianças com idade entre 5 e 6 anos já desenvolviam as noções de matemática apresentadas no aplicativo através do livro Buriti Mirim e outras atividades lúdicas realizadas pela professora.

Para o ensaio de interação as crianças foram organizadas em duplas ou trios para realizarem as atividades, em cada dupla ou trio tinha pelo menos uma criança alfabetizada. Esta didática foi escolhida pela professora, pois as crianças já estão habituadas a este método. Inicialmente notou-se que as crianças que ainda estão em fase de alfabetização tiveram receio ao iniciar o jogo, tendo em vista as nomeações das atividades, e estas crianças escolhiam as atividades pelas imagens contidas nos botões.

Pode ser observado que durante a utilização da atividade Contar Colorindo, a maioria das crianças selecionavam todas as imagens para depois ler o enunciado e objetivar a resposta correta, também se percebeu que as crianças tinham dificuldade em dar continuidade após selecionar as imagens, então se achou necessário mudar a imagem utilizada no botão continuar, para um melhor entendimento das crianças. Os alunos que não sabiam ler tiveram o auxílio dos colegas para ler o enunciado.

A atividade Subtração, por ser uma operação matemática que poucos dominavam, foi uma das atividades em que a maioria das crianças demonstrou menos interesse. Utilizando inicialmente para conhecer a atividade, mas sem interesse em uma segunda utilização.

Algumas crianças ao escolherem a atividade Contar com Formas Geométricas tinham facilidade para entender o objetivo da atividade, outras após o auxílio da professora concluíam a atividade com êxito. Por ser uma atividade onde a instrução é feita a partir de uma imagem, para objetivar quais das formas geométricas estavam em destaque naquele momento, algumas crianças tiveram dificuldades em entender de que a imagem da instrução não deveria ser contada juntamente com a imagem que apresentava as formas geométricas.

Percebeu-se que as crianças após observar que as mensagens apresentadas na recompensa variavam de acordo com a pontuação, fez com que o número de erros diminuísse. Conclui-se que o aplicativo apresentou resultados satisfatórios mostrando-se como uma alternativa viável e de boa aceitação por parte das crianças.

6 | CONCLUSÕES E TRABALHOS FUTUROS

Esta aplicação oferece a alternativa da utilização da tecnologia tanto dentro de sala de aula quanto fora sendo necessário para a criança apenas um dispositivo *mobile* para que a utilização do aplicativo seja possível. Diferentemente dos outros aplicativos disponíveis gratuitamente no mercado, o *Genius Math* oferece variadas atividades e níveis de dificuldades, afim de que a criança se sinta desafiada a concluir as atividades.

Como trabalhos futuros pretende-se realizar novos ensaios de interação em escolas públicas e em escolas onde o projeto Buriti Mirim não seja empregado com a intenção de identificar uma possível adaptação facilitando ao software pelo mesmo utilizar-se como base um material já conhecido das crianças. Também poderão ser implementados novos exercícios nas atividades já existentes. Outra possibilidade em estudo é adaptar a aplicação também para *smartphones* com o sistema operacional *Windows phone*, para assim atingir um maior público. E para finalizar intenciona-se disponibilizar o aplicativo gratuitamente na *PlayStore*.

REFERÊNCIAS

- BRAGA, Juliana. **Objetos de Aprendizagem Volume 2: metodologia de desenvolvimento**. Santo André: UFABC (2015b). pesquisa. ufabc. edu. br/intera, 2015.
- DE SOUSA MONTEIRO, Bruno et al. **Youubi: Ambiente de aprendizagem ubíqua**. In: Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE). 2015. p. 111.
- EDITORA MODERNA. **Buriti Mirim 2**. Disponível em: <<http://www.moderna.com.br/main.geld=4028818B2E24D324012E3469E60A34AF&itemId=04F137CC45E1405088B38F5CBD8200F0>>. Acesso em: 22 Ago 2015.
- TAJRA, Sanmya Feitosa. **Informática na Educação: novas ferramentas pedagógicas para o professor na atualidade**. Érica, 2011.
- FOWLER, Martin. **UML Essencial: um breve guia para linguagem padrão**. Bookman editora, 2014.
- GEORGIEV, Tsvetozar; GEORGIEVA, Evgenia; SMRIKAROV, Angel. **M-learning-a New Stage of E-Learning**. In: International conference on computer systems and technologies-CompSysTech. 2004. p. 1-4
- GIL, Antonio Carlos. **Como elaborar projetos de pesquisa**. São Paulo, v. 5, n. 61, p. 16-17, 2002.
- HRASTINSKI, Stefan. **Asynchronous and synchronous e-learning**. Educause quarterly, v. 31, n. 4, p. 51-55, 2008.
- IBGE. **Acesso à Internet e à televisão e posse de telefone móvel celular para uso pessoal**. Disponível em: <<http://biblioteca.ibge.gov.br/visualizacao/livros/liv93373.pdf>>. Acesso em: 10 Abr 2015.
- NIELSEN, Jakob. **Designing web usability: The practice of simplicity**. New Riders Publishing,

1999.

NORTVIG, A. **The change of time and space in e-learning**. American Journal of Educational Research, v. 2, n. 8, p. 612-616, 2014.

PRESSMAN, Roger S. **Engenharia de software: uma abordagem profissional**. 7ª Edição. Ed: McGraw Hill, 2011.

SANTOS, J. O. **O lúdico na Educação Infantil**. Campina Grande: Realize, 2011.

WELLER, Martin; PEGLER, Chris; MASON, Robin. **Putting the pieces together: What working with learning objects means for the educator**. In: the Proceedings of the Second eLearnInternational World Summit, Edinburgh International Conference Centre, Edinburgh, Scotland. 2003.

APLICATIVO MÓVEL PARA PREPARAÇÃO DE ESTUDANTES PARA O ENEM NO CONTEXTO DA DISCIPLINA DE MATEMÁTICA

Hannderson Faria Arantes

Universidade Federal de Itajubá, Instituto de Matemática e Computação
Itajubá – Minas Gerais

Rodrigo Duarte Seabra

Universidade Federal de Itajubá, Instituto de Matemática e Computação
Itajubá – Minas Gerais

RESUMO O ENEM (Exame Nacional do Ensino Médio) é uma avaliação elaborada pelo Ministério da Educação e tem como finalidade mensurar o domínio de competências e habilidades de estudantes que estão concluindo o Ensino Médio. Atualmente, o exame é amplamente utilizado como instrumento de acesso pela maioria das instituições públicas do país. Assim, a preparação dos estudantes é essencial para a obtenção de bons resultados durante a disputa por uma vaga no ensino superior. Esta pesquisa apresenta o desenvolvimento de um aplicativo para a plataforma Android visando auxiliar a preparação de estudantes para o ENEM, no contexto da disciplina de matemática. Os principais resultados mostraram que a ferramenta se apresenta como uma boa alternativa a ser explorada em sala de aula.

PALAVRAS-CHAVE Aplicativos móveis; ENEM; Matemática; TME; Ensino.

ABSTRACT The ENEM (National Examination of Secondary Education) is an assessment developed by the Ministry of Education and aims to measure the domain of competences and skills of students who are completing high school. Currently, the test is widely used as a tool of access to the majority of public institutions of the country. Thus, the preparation of students is essential for obtaining good results during the competition for a place at higher education. This research presents the development of an application for the Android platform in order to assist the preparation of students for the ENEM in the context of the mathematics discipline. The main results showed that the tool is presented as a good alternative to be explored in the classroom.

KEYWORDS Mobile applications; ENEM; Mathematics; TME; Learning.

1 | INTRODUÇÃO

Atualmente, vivencia-se uma árdua concorrência quanto ao ingresso de estudantes em uma instituição de ensino superior no Brasil. Por conta disso, faz-se necessária uma preparação intensa do estudante para que esteja apto a conquistar sua vaga em uma universidade.

Com o objetivo de avaliar estudantes no final do Ensino Médio, em 1998, foi criado o Exame Nacional do Ensino Médio (ENEM). Na atualidade, o exame é o principal meio de ingresso de estudantes em universidades brasileiras, sobretudo no caso das instituições públicas. Segundo dados do Ministério da Educação (MEC 2015), o número de instituições que aderiram ao ENEM já ultrapassa 120 unidades.

Aliado a esse cenário, é cada vez mais perceptível que novos paradigmas vêm sendo aplicados no processo de ensino-aprendizagem, visando melhorar a compreensão dos assuntos estudados dentro e fora da sala de aula. Em relação a esse aspecto, aplicativos educacionais gratuitos estão à disposição de docentes e discentes por meio da Internet, contribuindo para um ambiente de aprendizado mais diversificado e interessante.

Do ponto de vista de Marçal, Andrade e Rios (2005), com a ampla disseminação do uso de dispositivos móveis, notou-se que esta tecnologia poderia ser empregada na educação como parte de um modelo de aprendizado integrado. O uso de dispositivos móveis na educação criou um novo conceito, denominado *mobile learning*, ou, apenas, *m-learning*. Já em 2002, Nyiri (2002) afirma que o paradigma *m-learning* surge da intenção de aproveitar a disponibilidade de dispositivos móveis considerando as necessidades específicas de educação e treinamento. De modo complementar a essa premissa, Rheingold (2007) ressalta que a ubiquidade dos dispositivos móveis conduzirá o *m-learning* como um meio importante para fornecer educação e informação. O poder ubíquo dos dispositivos móveis para se aprender em qualquer lugar e a qualquer momento tem conquistado cada vez mais adeptos que buscam não só complementar o ensino, mas realizar estudos nos mais diversos locais.

McGreal (2005) afirma que a aprendizagem concebida por meio do *m-learning* terá um impacto significativo na aprendizagem tradicional. Para este autor, num futuro próximo, esse tipo de aprendizagem se tornará parte habitual da educação de todos e complementar o contexto de aprendizagem da sociedade, levando este paradigma a ganhar pertinência e visibilidade.

A partir do cenário exposto, esta pesquisa tem como objetivo central apresentar o desenvolvimento de um aplicativo que privilegie a compreensão e o treinamento de conceitos matemáticos por meio de um aplicativo *mobile* gratuito disponível na plataforma *Android*. Espera-se, portanto, que o aplicativo educacional proposto auxilie os estudantes em sua preparação para a avaliação de matemática do ENEM, abordando conteúdos relacionados ao ensino desta disciplina. Adicionalmente, o trabalho ainda se propõe a investigar, sob a ótica discente, os potenciais benefícios apontados em decorrência do uso do aplicativo proposto no apoio ao estudo da matemática. Como principais contribuições do estudo destacam-se: (i) disseminar a ideia do uso de aplicativos móveis, para que futuramente haja aumento de sua utilização no âmbito educacional; (ii) avaliar a opinião dos estudantes envolvidos no estudo, verificando se o aplicativo desenvolvido nesta oportunidade atendeu às expectativas esperadas, bem como seu grau de aceitação e contribuição no processo de aprendizagem.

A seção 2 discute a importância do ENEM no cenário educacional, o uso de aplicativos *m-learning* no ensino, bem como alguns trabalhos correlatos. A seção 3 discorre sobre a apresentação do aplicativo desenvolvido neste estudo e o método empregado em sua avaliação. A seção 4 apresenta a análise dos dados coletados na pesquisa experimental realizada e, finalmente, a seção 5 apresenta as conclusões do trabalho.

2 | FUNDAMENTAÇÃO TEÓRICA

O ENEM, criado pelo Instituto Nacional de Pesquisas Educacionais Anísio Teixeira (INEP), do Ministério da Educação (MEC), tem como finalidade avaliar a qualidade do Ensino Médio no país, bem como servir como critério de seleção para que estudantes ingressem no Ensino Superior. A avaliação é composta por uma redação e provas objetivas que avaliam as seguintes áreas de conhecimento: Linguagens, Códigos e suas Tecnologias, Ciências Humanas e suas Tecnologias, Ciências da Natureza e suas Tecnologias, e Matemática e suas Tecnologias (MEC 2015).

Desde a sua criação, o ENEM vem constituindo uma área de interesse de estudo tanto no âmbito político quanto no educacional, como pode ser visto nos trabalhos de Lopes e López [6], com um estudo sobre a performatividade nas políticas de currículo no caso do ENEM, e de Castro e Tiezzi (2004), com uma pesquisa sobre a reforma do Ensino Médio e a implantação do ENEM no Brasil. Para Silva e Ribas (2009), o ENEM se difere de outros exames propostos pelo MEC, pois, além de centrar-se em avaliar o desempenho do estudante por competências e habilidades, privilegia a interdisciplinaridade e a contextualização.

De acordo com Castro e Tiezzi (2004), o que está presente na concepção do ENEM é a importância de uma educação com conteúdos mais ricos, voltada para o desenvolvimento da capacidade cognitiva e de raciocínio dos estudantes. A formação não está mais pautada na memorização maciça de informações e fatos, mas no desenvolvimento de estruturas mentais que permitem ao jovem e ao adulto enfrentar problemas novos, usando as tradicionais teorias científicas.

Como uma das áreas do conhecimento abrangidas pelo ENEM, segundo Ponte *et al* (2007), a matemática é uma das ciências mais antigas e uma das mais tradicionais disciplinas escolares, tendo sempre ocupado um lugar relevante no currículo. Para Cazorla (2012), diversos indivíduos afirmam possuir aversão a essa disciplina e, para a maioria, isso acontece por lembranças de aprendizagens escolares. Uma possível explicação para isso se deve à falta de percepção das possíveis aplicações sobre o conhecimento estudado, bem como sua utilidade para a vida desses indivíduos. Ademais, de acordo com Carvalho (2009) e Ponte e Sousa (2010), o fato de as aulas de matemática em nível de primeiro, segundo e terceiro graus ainda serem realizadas

de forma expositiva, dificulta o desenvolvimento cognitivo do estudante. O aprendiz passa a realizar exercícios de aplicação, que nada mais são do que uma repetição da aplicação de um modelo de solução apresentado pelo docente.

Em oposição a essa prática, Marçal, Andrade e Rios (2005) defendem que os conceitos de *m-learning* favorecem um aprendizado dinâmico, beneficiando tanto trabalhadores como estudantes, que enfrentam longos deslocamentos por suas rotinas envolverem viagens ou traslados para diferentes locais. A preocupação é fornecer um ambiente de aprendizado que proporcione a informação mais atualizada possível, bem como o suporte tanto para a aprendizagem formal como para a informal, fomentando meios para o desenvolvimento de métodos inovadores de ensino utilizando recursos da computação e da mobilidade. Desta forma, o *m-learning* surge como uma importante ferramenta alternativa de ensino, aprendizagem e treinamento à distância.

2.1 Trabalhos correlatos

Batista, Behar e Passerino (2010), com o objetivo de promover uma classificação de aplicativos e objetos de aprendizagem sobre dispositivos móveis direcionados à matemática, realizaram uma análise criteriosa a respeito de alguns *softwares* comerciais e gratuitos. Foi constatado que a popularização dos dispositivos móveis é um aspecto positivo em termos educacionais, uma vez que favorece o alcance de um grande número de indivíduos, sem requerer deslocamentos físicos. A pesquisa também destacou que há necessidade de inovações no processo de ensino-aprendizagem no contexto da matemática, pois ainda são poucas as iniciativas que consideram o uso de dispositivos móveis como ferramenta de apoio.

No trabalho de Neto e Fonseca (2013), com o intuito de estimular o aprendizado da matemática utilizando jogos educativos para dispositivos móveis, foi desenvolvido um jogo, para a plataforma *Android*, baseado na obra literária de Malba Tahan: “O homem que calculava”. Visando comprovar a efetividade de sua proposta, os resultados apontaram que a maioria dos estudantes aprovou o apoio da mobilidade no uso de jogos educativos digitais, devido à flexibilidade de uso independente do local.

Com o objetivo de apresentar uma elicitación dos principais requisitos do desenvolvimento de aplicações *m-learning*, sobretudo as destinadas ao ensino de matemática, Marçal *et al* (2010) definiram uma base para a criação de um guia prático para este tipo de aplicação. Na ocasião, foram desenvolvidos três aplicativos no âmbito educacional: M-Queops, M-Prisma e M-Pitagoras. Os resultados da pesquisa apontaram que o emprego de dispositivos móveis se apresenta como uma interessante ferramenta de auxílio ao ensino e à aprendizagem. Além disso, foi constatado que o estudo de conceitos matemáticos por meio de *m-learning* apresenta algumas particularidades que exigem desde a adaptação de conteúdos às pequenas telas até a adoção de métodos pedagógicos específicos.

Na pesquisa realizada por Batista, Behar e Passerino (2011), foi proposto um

modelo pedagógico, denominado *m-LearnMat*, visando a orientar atividades de *m-learning* em matemática com foco no ensino superior. O modelo foi aplicado em turmas de Cálculo I de uma Instituição Federal de Ensino utilizando o celular como dispositivo móvel. Foi constatado que o modelo pedagógico proposto orientou a proposta da disciplina, que foi fortemente apoiada pelas tecnologias digitais.

O estudo realizado por Sena, Oliveira e Carvalho (2014) teve como objetivo apresentar um mapeamento sistemático da literatura sobre aplicações *m-learning* utilizadas no ensino de matemática, com conteúdo e recursos diferenciados e aplicados em diferentes níveis de ensino (Fundamental, Médio e Superior). Os resultados apontaram a importância de se analisar o impacto dos aplicativos *m-learning*, pois eles podem influenciar a motivação e o desempenho dos estudantes. Ademais, foi constatado que há necessidade de se promover inovações no processo de ensino-aprendizagem no contexto de matemática. Os autores também destacaram que ainda são poucas as iniciativas que consideram o uso de dispositivos móveis como ferramenta de apoio.

A partir das pesquisas arroladas, é fato que estudos envolvendo o uso de aplicativos móveis têm despertado o interesse de pesquisadores no desenvolvimento de projetos diferenciados que facilitem o ensino e a aprendizagem de matemática. Na mesma direção dos estudos apresentados, o fator motivador envolvido nesta oportunidade também consiste em proporcionar apoio aos estudantes que, muitas vezes, demonstram ou relatam algum tipo de dificuldade na aprendizagem desta disciplina. Deste modo, o presente trabalho se apresenta como mais uma alternativa frente a esse desafio, abrangendo uma nova oportunidade para o treinamento de conceitos matemáticos como forma de preparação para o ENEM.

3 | MÉTODO

3.1 O Aplicativo TME

O TME (Treinamento de Matemática para o ENEM) consiste em uma aplicação *Android* com a finalidade de fornecer apoio ao processo de treinamento para o ENEM na disciplina de matemática. A seleção da plataforma foi realizada de forma a abranger a maior quantidade possível de usuários. Para o desenvolvimento da aplicação, foi utilizada a IDE *Android Studio*, concebida pelo Google, que consiste em um ambiente para a criação de aplicações para a plataforma *Android*. O desenvolvimento ocorreu utilizando a linguagem Java e XML. Para o armazenamento dos dados, foi utilizado o banco de dados SQLite. Vale ressaltar que o TME é totalmente gratuito.

Ao iniciar o processo de interação com o TME e antes de o usuário obter acesso ao *menu* principal, é necessário inserir uma conta do Google para manter as estatísticas a serem exibidas ao usuário. Essa restrição se deve à facilidade em estabelecer este

vínculo, tendo em vista que todo usuário do sistema operacional *Android* deve possuir uma conta Google para acesso a *Play Store*, local no qual o usuário realiza o *download* de aplicações. O aplicativo, quando iniciado, estabelece uma conexão com a conta Google automaticamente, caso o dispositivo esteja conectado à internet. O vínculo do aplicativo com a conta Google oferece algumas vantagens, por exemplo, a manutenção automática de um *ranking* localizado na *Play Games*, que pode ser acessado por duas modalidades: social e todos. No primeiro caso, o *ranking* é composto apenas por pessoas que estão na lista de contatos do usuário do aplicativo TME. No segundo, há um *ranking* geral composto por todos os usuários do aplicativo.

Outro aspecto a ser ressaltado é a possibilidade de “conquistas” serem atribuídas com base em determinados feitos que o usuário consiga realizar durante o uso do aplicativo. As conquistas *acumulativas* mantêm um registro de desempenho desde o primeiro uso do aplicativo e, à medida que o usuário progride, sua porcentagem aumenta até que a próxima conquista seja desbloqueada. As conquistas *diretas* não armazenam o progresso entre os usos, portanto, a cada novo uso o progresso do usuário é reiniciado. Para desbloquear esse tipo de conquista, o usuário deverá completá-la em apenas um uso. As conquistas têm a finalidade de estimular o uso do aplicativo por seus usuários, pois, a cada conquista desbloqueada, os jogadores são recompensados com pontos de experiência adicionados em seu perfil de conta Google.

O usuário pode selecionar o nível das questões que deseja responder, bem como ter acesso às suas estatísticas de respostas, a saber: quantidade de acertos e erros em cada nível (fácil, médio ou difícil) e números totais de tentativas. A Figura 1 ilustra telas de cada um dos diferentes níveis de questões em um *smartphone*.

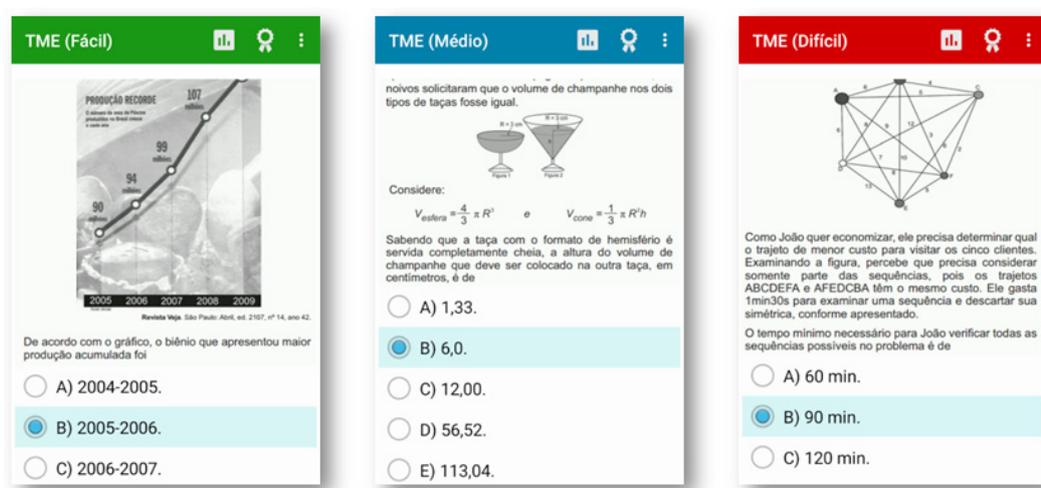


Figura 1. Exemplos de questões com base nos diferentes níveis.

Fonte: Os autores.

A interface foi projetada para atuar de forma responsiva, visando melhor experiência do usuário independentemente do aparelho que estiver executando o

aplicativo. Essa escolha se deve à grande variedade de dimensões das telas dos dispositivos atualmente, e principalmente em razão da diferença entre *tablets* e *smartphones*. Existe, ainda, a possibilidade de o usuário aplicar efeito de *zoom* no enunciado das questões, facilitando a leitura em dispositivos menores.

Acada questão respondida, uma mensagem notificando o acerto ou erro é exibida para o usuário. Em ambos os casos, a resolução passo a passo da respectiva questão é apresentada, fornecendo explicações adicionais para o esclarecimento de dúvidas. Essa decisão se deve ao fato de o ENEM ser caracterizado como um exame que testa não só os conhecimentos do estudante, mas sua resistência. Com isso, existe a possibilidade de o aluno apresentar dúvida e acabar optando pela resposta correta, mesmo sem ter o conhecimento requerido. Por esse motivo, mesmo em caso de acertos, a resolução da questão é exibida. A Figura 2 ilustra exemplos de mensagens de erro e acerto, o *ranking* dos usuários do aplicativo e estatísticas de respostas.

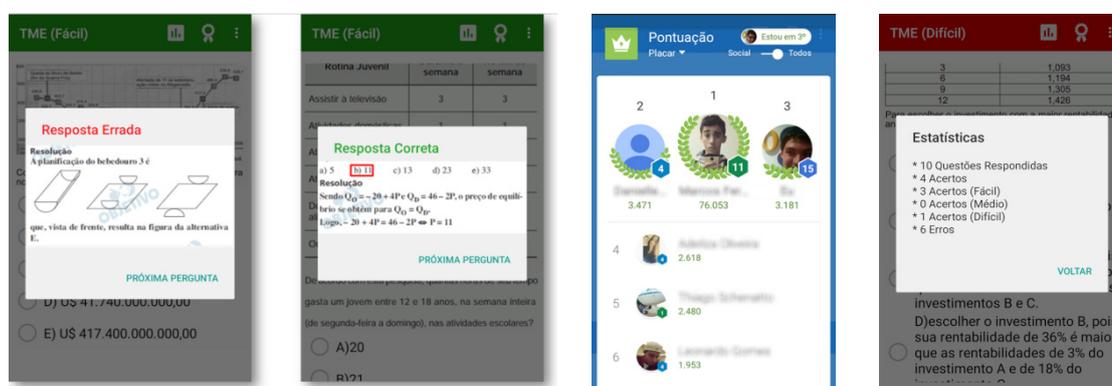


Figura 2. Exemplos de erro/acerto de questão, placar geral e estatísticas do aplicativo.

Fonte: Os autores.

3.2 Participantes e descrição do método

Os estudantes selecionados para participarem do estudo eram correspondentes ao 3º ano do Ensino Médio e de um curso Pré-Vestibular (PV), com idade média de 18 anos. A pesquisa experimental foi realizada em uma instituição privada de ensino da cidade de Itajubá-MG. Inicialmente, foram selecionadas cinco turmas, que foram separadas em dois grupos: turma A, que representa os alunos do 3º ano, e turma B, que corresponde aos alunos do PV. Os dois grupos juntos totalizaram 109 alunos presentes no primeiro encontro, que envolveu a apresentação do aplicativo. No entanto, somente parte deles utilizou de fato a aplicação no período disponível para experimentação e, por conseguinte, puderam participar da segunda etapa, que consistiu em responder um questionário avaliativo sobre o uso do aplicativo TME. A turma A foi composta por 15 homens e nove mulheres; a turma B por quatro homens e três mulheres, totalizando 31 estudantes participantes da pesquisa experimental.

O método para a aplicação da pesquisa experimental consistiu de duas etapas.

Durante a primeira etapa, foram realizadas duas apresentações sobre a aplicação desenvolvida para os estudantes de ambas as turmas (A e B), com duração aproximada de 10 minutos cada. Para a realização da apresentação, foi preparado um breve material que foi exposto aos alunos com o intuito de demonstrar o aplicativo. Após o término das apresentações, os estudantes foram convidados a utilizar o aplicativo disponível gratuitamente na *Play Store*. A segunda etapa foi realizada um mês após a primeira. Esse intervalo de tempo foi necessário para que os alunos pudessem utilizar o aplicativo.

Após este período, foi realizada a aplicação de um questionário avaliativo sobre o uso experimentado do aplicativo pelos estudantes. O questionário consistiu em 10 questões de múltipla escolha, cujo objetivo foi minimizar a subjetividade de questões abertas, além de oferecer um espaço para a descrição de aspectos relevantes caso os alunos desejassem manifestar comentários adicionais. Durante a aplicação do questionário, os autores da pesquisa não interferiram ou ajudaram os participantes em respondê-lo. Os participantes foram informados sobre o objetivo da atividade a qual foram submetidos, bem como sobre o anonimato e uso dos dados coletados apenas para fins de pesquisa. O Quadro 1 apresenta as questões investigadas, além de uma questão final que solicita comentários de ordem geral.

Questões
Q1. Com que frequência semanal você fez uso do aplicativo?
Q2. De maneira geral, a utilização do aplicativo foi de fácil compreensão e assimilação?
Q3. Os recursos visuais disponibilizados pelo aplicativo (cores, ícones e imagens) são de boa qualidade?
Q4. Você acha que o uso do aplicativo contribuiu para um treinamento adequado para o ENEM?
Q5. Você acha que o uso do aplicativo contribuiu para auxiliar a aprendizagem de matemática?
Q6. Você gostaria que o aplicativo tivesse suporte a outras disciplinas, tais como português, inglês, redação, história, geografia etc.?
Q7. De maneira geral, como você avalia a performance do aplicativo no seu <i>smartphone</i> ?
Q8. A qualidade e a quantidade de questões disponíveis no aplicativo foram adequadas?
Q9. O espaço necessário para o armazenamento do aplicativo no <i>smartphone</i> foi adequado?
Q10. O aplicativo é de fácil instalação e desinstalação?
Comentários adicionais:

Quadro 1. Questões da avaliação sobre o uso do aplicativo.

4 | ANÁLISE DOS DADOS

A primeira questão da avaliação buscou identificar a frequência de uso do aplicativo durante o intervalo entre as duas etapas. De modo geral, 62% dos participantes utilizaram o TME pelo menos três vezes na semana (Figura 3). A participação do público masculino foi um pouco maior, com uso médio semanal aproximado de 3,5 vezes. No caso das mulheres, o uso semanal foi de 2,5 vezes, aproximadamente. Em relação às turmas, o uso médio foi diferente. A turma A (3º ano) realizou um uso mais frequente, alcançando uma média de 3,7 vezes na semana. Já a turma B, utilizou apenas 1,5 vezes na semana.

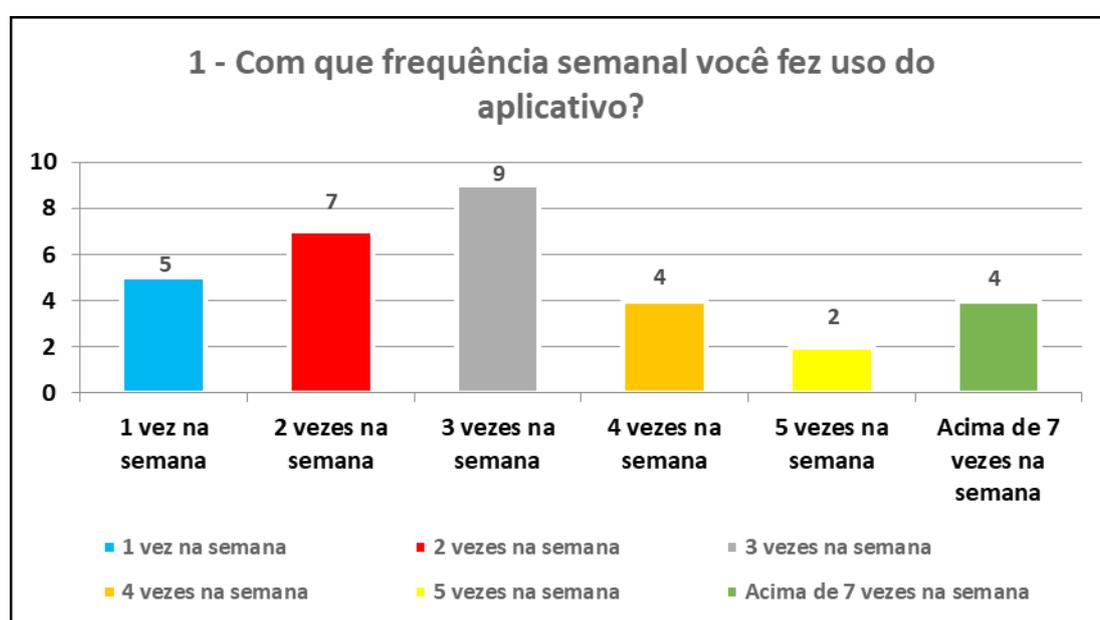


Figura 3. Frequência de uso do aplicativo.

Fonte: Os autores.

A segunda questão está relacionada à usabilidade do aplicativo. Nesse contexto, 100% (71% atribuíram *totalmente* e 29% responderam *muito*) dos participantes afirmaram que a ferramenta apresenta boa facilidade de uso. Esse resultado é ilustrado na Figura 4. Considerando os gêneros, 84,6% do público feminino avaliou a usabilidade como *totalmente* adequada (15,4% responderam *muito*), enquanto que, para os homens, este índice foi de 63,2% (36,8% responderam *muito*). Considerando cada turma, ambas apontaram 100% de aprovação, sendo esta porcentagem composta por 70,8% dos alunos que estavam *totalmente* de acordo na turma A. Na turma B, 57,1% dos participantes manifestaram essa mesma opinião (*totalmente*).

A próxima investigação (questão 3) diz respeito ao aplicativo oferecer os recursos visuais necessários para uma boa interação segundo a visão dos estudantes. Neste caso, 97% (68% responderam *excelente* e 29% atribuíram *bom*) dos participantes concordaram que os recursos visuais são de boa qualidade. Somente 3% dos

participantes apontaram essa resposta como *regular* (Figura 5). A aprovação do público feminino foi maior, pois 84,6% das mulheres relataram que os recursos são *excelentes*. Para os homens, este número corresponde a 57,9% (36,8% responderam *bom*). Para as turmas, foi obtida uma aceitação *excelente* de 62,5% por parte do 3º ano (37,5% responderam *bom*); já na turma do PV, 71,4% dos participantes manifestaram opinião *excelente* (14,3% afirmaram *bom*).

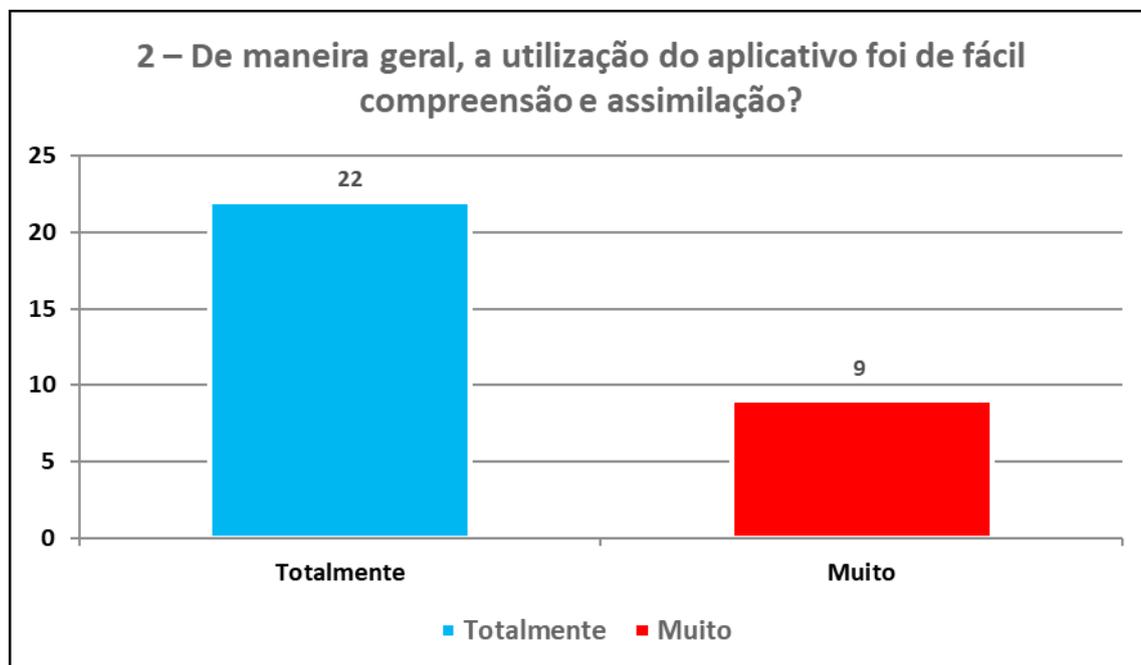


Figura 4. Recursos de usabilidade do aplicativo.

Fonte: Os autores.

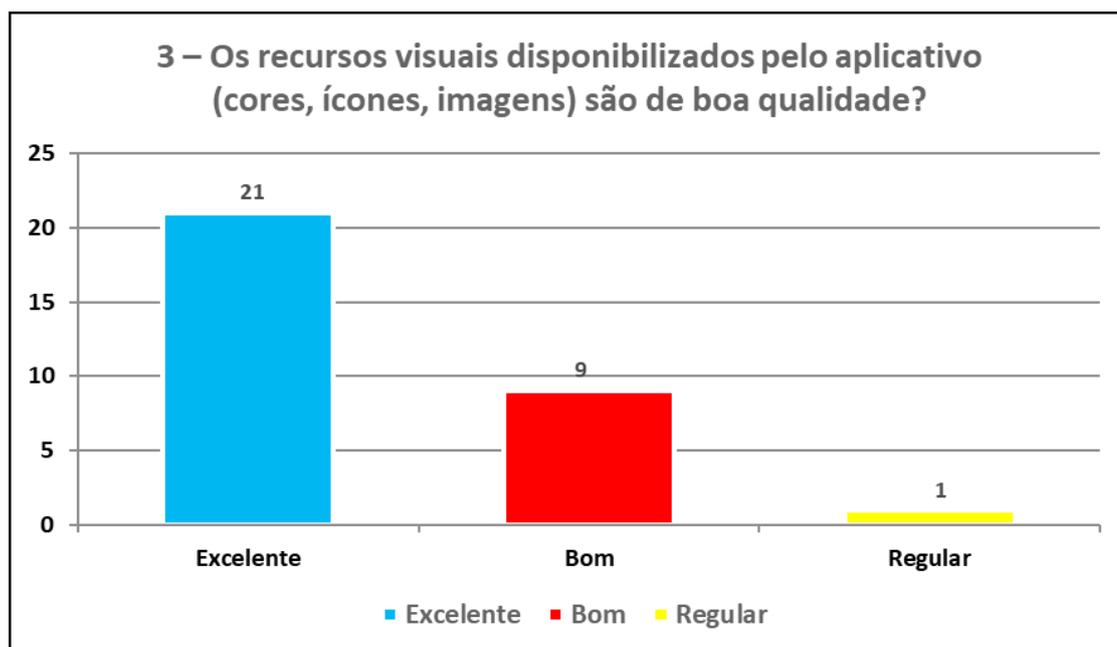


Figura 5. Recursos visuais do aplicativo.

Fonte: Os autores.

A quarta questão visou determinar se, de fato, o aplicativo cumpriu com seu propósito de auxiliar no estudo para o ENEM. Foi possível constatar que 93% dos estudantes classificaram essa contribuição como *totalmente* (48%) ou *muito* (45%), enquanto apenas 7% classificaram como *regular* (Figura 6). Essa opinião neutra pode ser atribuída ao fato de que alguns alunos apontaram a falta do monitoramento do tempo durante o uso do TME como um recurso essencial ao treinamento. De acordo com o sexo, a avaliação feminina foi 100% positiva (46,2% *totalmente* e 53,8% *muito*), enquanto 90% dos homens foram favoráveis (47,4% *totalmente* e 42,1% *muito*). Analisando as turmas, os alunos do 3º ano foram unânimes quanto à aprovação (50% *totalmente* e 50% *muito*). Para a turma do PV, 71,4% dos participantes aprovaram a contribuição, porém 28,5% apontaram opinião *regular* nesta questão. Provavelmente, o fato de os alunos do PV terem usado o TME com menor regularidade (questão 1) tenha refletido nessa resposta.

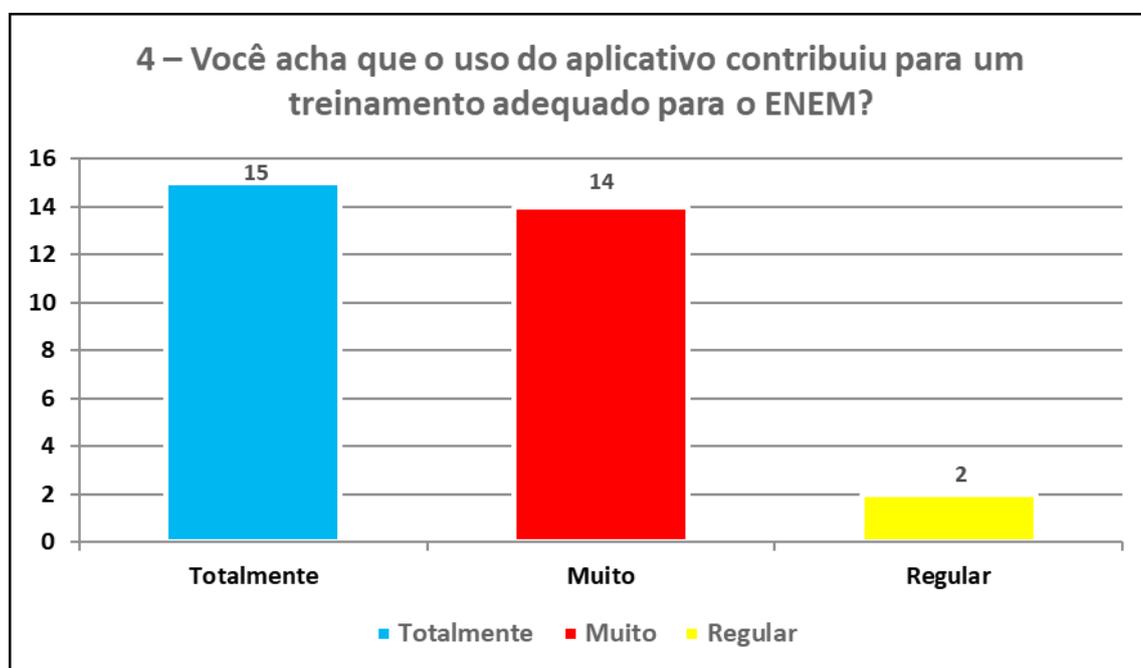


Figura 6. Contribuição do aplicativo para o treinamento do ENEM.

Fonte: Os autores.

Em relação à contribuição para o aprendizado de matemática, 75% dos estudantes manifestaram opinião favorável nesta questão, os demais afirmaram que a contribuição foi *regular* ou *pouca*. Esse índice pode ser justificado pelo fato de que o aplicativo não oferece materiais para ensinar o conteúdo de matemática, mas sim exercícios que visam treinar o conteúdo previamente estudado. Nesta questão, 73,7% dos homens manifestaram opiniões positivas, enquanto que, para as mulheres, esta porcentagem foi de 77%. Apesar de 83% da turma do 3º ano ter opinado favoravelmente em relação a essa questão, 42,9% dos estudantes da turma do PV se mostraram neutros, afirmando que a contribuição foi *regular*, e 14,3% manifestaram *pouco* como resposta.

Novamente, esses resultados podem ser justificados pelas respostas da questão 1.

A sexta questão foi elaborada com o objetivo de verificar a aceitação desse tipo de aplicação para o aprendizado. De acordo com os resultados, 100% dos estudantes se mostraram favoráveis a esta iniciativa. Ademais, alguns participantes sugeriram que os autores deveriam expandir o aplicativo para outras disciplinas. Em relação aos gêneros e às turmas, a aprovação foi unânime.

A sétima questão – “De maneira geral, como você avalia a performance do aplicativo no seu smartphone?” – teve como objetivo avaliar se o aplicativo se comportou de maneira adequada nos mais diversos dispositivos. Neste caso, 90% dos estudantes manifestaram que o desempenho foi *excelente* (61%) ou *bom* (29%), enquanto os demais o classificaram como *regular* (Figura 7). As opiniões neutras podem ser explicadas devido à grande variedade de dispositivos disponíveis no mercado atualmente, com diferentes capacidades de processamento. Provavelmente, o desempenho tenha sido regular em determinados modelos.

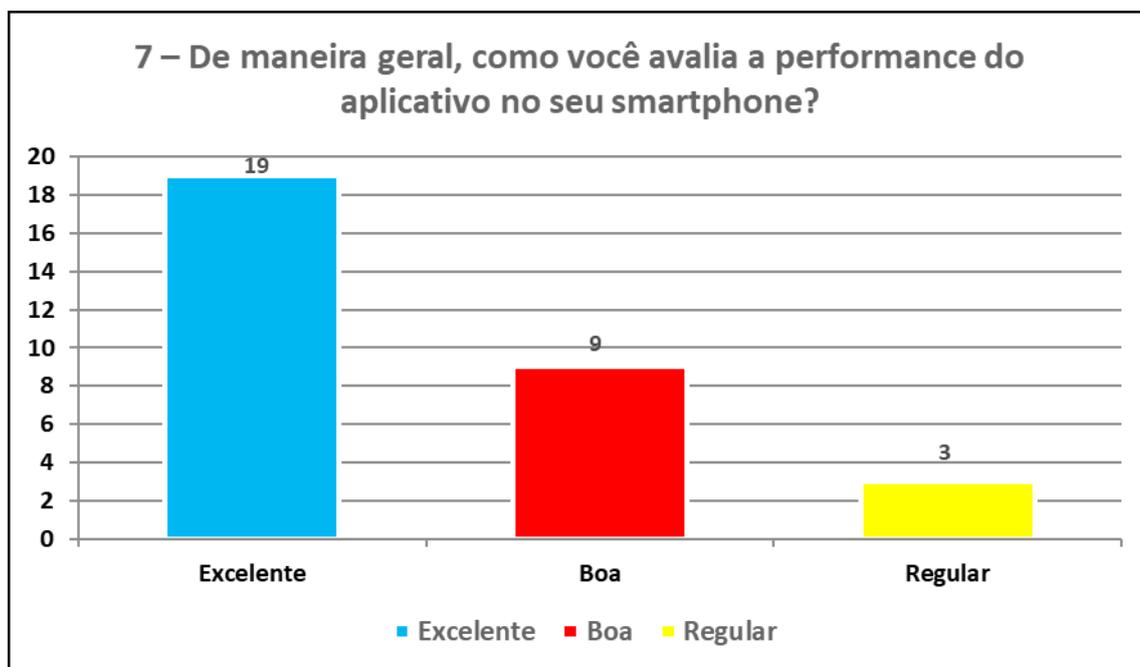


Figura 7. Desempenho do aplicativo.

Fonte: Os autores.

No caso da oitava questão, 87% dos participantes afirmaram que o aplicativo está *totalmente* ou *muito* de acordo com a qualidade e a quantidade de questões presentes, enquanto 13% dos estudantes classificaram esses quesitos como *regular* ou *pouco*. De acordo com os sexos, 100% das mulheres estão *totalmente* ou *muito* de acordo. Para os homens, esta porcentagem alcançou 78,9% das opiniões. Em relação às turmas, os estudantes do PV foram unânimes quanto à aprovação. A turma do 3º ano foi favorável com 83,3% das respostas.

No que compete à questão 9, 74% dos participantes ficaram satisfeitos com o

tamanho de armazenamento do aplicativo. Dentre aqueles que manifestaram opinião negativa (apenas 3%), este posicionamento pode ser explicado pela grande variedade de tamanho das memórias dos *smartphones* atualmente. Para os usuários que possuem certa limitação deste recurso, o armazenamento do aplicativo pode não ter sido adequado em função deste fato. Em relação aos sexos e às turmas, a situação foi semelhante.

A última questão investigada comprova a facilidade para instalação do aplicativo. O TME é compatível com diversas versões do *Android*, abrangendo qualquer versão acima da 2.3.3. A instalação se torna possível por meio da *Play Store*, na qual o aplicativo está disponível de forma gratuita para *download*. Neste âmbito, 100% dos estudantes alegaram não haver dificuldades para instalar ou desinstalar o aplicativo. Este é um fator de suma importância, visto que a partir da facilidade de acesso ao aplicativo, sua utilização torna-se mais atrativa no contexto educacional, sobretudo se incorporando mais presente no dia a dia dos estudantes. Analisando os gêneros e as turmas, o resultado foi o mesmo.

5 | CONSIDERAÇÕES FINAIS

Os aplicativos são ferramentas poderosas capazes de realizar as mais diversas tarefas. Contudo, recorrer a aplicativos pagos pode tornar seu uso oneroso e, desta forma, inviável. Os aplicativos gratuitos permitem democratizar o acesso ao conhecimento e à informação.

Com o avanço e o baixo custo de acesso à tecnologia, os *smartphones* se difundiram pela sociedade. Em virtude disso, centenas de aplicativos surgem diariamente. A categoria de aplicações educacionais está em constante expansão e, atualmente, existem diversos projetos de aplicativos educacionais gratuitos desenvolvidos para atender desde a educação infantil até a superior. Nesta pesquisa, foi concebido o TME, que é um aplicativo direcionado ao treinamento da disciplina de matemática para o ENEM.

Os resultados observados a partir da avaliação da ferramenta constataam que, de modo geral, houve aprovação dos estudantes com relação ao uso do TME. Os comentários relatados pelos participantes da pesquisa corroboram o resultado por meio de inúmeras opiniões positivas. A maioria dos estudantes alegou que o aplicativo proporcionou apoio adequado ao complementar o treinamento na disciplina de matemática para o ENEM. Além disso, as estatísticas geradas pelo *Google Play* apontam que o aplicativo atualmente conta com 89.297 *downloads*.

Verifica-se, assim, que os aplicativos educacionais têm grande potencial para complementar o processo de aprendizagem. Porém, a falta de iniciativas na criação de aplicativos desta natureza e a mudança cultural de muitos docentes ainda são os

principais desafios encontrados com relação a este tema.

REFERÊNCIAS

- Batista, S. C. F.; Behar, P. A.; Passerino, L. M. (2010). Recursos pedagógicos para dispositivos móveis: uma análise com foco na matemática. **RENOTE: Revista Novas Tecnologias na Educação**, v. 8, n. 3.
- Batista, S. C. F.; Behar, P. A.; Passerino, L. M. (2011). M-learnMat: aplicação de um modelo pedagógico para atividades de m-learning em matemática. **Anais do XXII Simpósio Brasileiro de Informática na Educação – SBIE**, Aracaju.
- Carvalho, D. L. (2009). **Metodologia do ensino da matemática**. 3. Ed. São Paulo: Cortez Editora.
- Castro, M. H. G.; Tiezzi, S. (2004). A reforma do ensino médio e a implantação do ENEM no Brasil. **Desafios**, v. 65, n. 11, p. 46-115.
- Cazorla, I. M. (2012). **Metodologia do ensino da matemática**. 3. Ed. Bahia: Editus Editora.
- Lopes, A. C.; López, S. B. (2010). A performatividade nas políticas de currículo: o caso do ENEM. **Educação em revista**, v. 26, n. 1, p. 89-110.
- Marçal, E. et al. (2010). Da elicitação de requisitos ao desenvolvimento de aplicações de mobile learning em matemática. **Anais do XXI Simpósio Brasileiro de Informática na Educação – SBIE**, João Pessoa.
- Marçal, E.; Andrade, R.; Rios, R. (2005). Aprendizagem utilizando dispositivos móveis com sistemas de realidade virtual. **RENOTE: Revista Novas Tecnologias na Educação**, v. 3, n. 1.
- McGreal, R. (2005). Mobile devices and the future of free education. **Proceedings of ICDE World Conference, International Council for Distance Education**.
- MEC. Ministério da Educação. (2015). **Portal do MEC: Legislação**. Brasília. Disponível em: <<http://portal.mec.gov.br/index.php>>
- Neto, J. F. B.; Fonseca, F. S. (2013). Jogos educativos em dispositivos móveis como auxílio ao ensino da matemática. **RENOTE: Revista Novas Tecnologias na Educação**, v. 11, n. 1.
- Nyiri, K. (2002). Towards a philosophy of m-learning. **Proceedings of IEEE International Workshop on Wireless and Mobile Technologies in Education – WMTE**.
- Ponte, J. P. et al. (2007). **Programa de matemática do ensino básico**. Ministério da Educação.
- Ponte, J. P.; Souza, H. (2010). **Uma oportunidade de mudança na matemática no ensino básico**. Lisboa: APM.
- Rheingold, H. (2007). **Smart mobs: the next social revolution**. Basic Books.
- Sena, D. M.; Oliveira, E. H.; Carvalho, L. S. (2014). Aplicativos móveis para o aprendizado de matemática. **Anais do XXV Simpósio Brasileiro de Informática na Educação – SBIE**, Dourados.
- Silva, E. F.; Ribas, M. H. (2009). A prova do ENEM: o que pensam os professores de matemática? **Olhar de Professor**, v. 6, n. 1.

COTIDIANO: UM SOFTWARE PARA AUXILIAR CRIANÇAS AUTISTAS EM SUAS ATIVIDADES DIÁRIAS

Afranio Furtado de Oliveira Neto
Instituto Federal do Triângulo Mineiro
Uberaba – MG

Hugo Leonardo Pereira Rufino
Instituto Federal do Triângulo Mineiro
Uberaba – MG

Diovane de Godoi Beira
Instituto Federal do Triângulo Mineiro
Uberaba – MG

Rodolfo Bocado Palis
Instituto Federal do Triângulo Mineiro
Uberaba – MG

Paula Teixeira Nakamoto
Instituto Federal do Triângulo Mineiro
Uberaba – MG

essas dificuldades, foi realizado um estudo para averiguar as tecnologias existentes que auxiliam no tratamento do autismo, logo foi possível observar a escassez de softwares de qualidade que levam em consideração essas premissas. Diante disso, foi realizado um estudo para o desenvolvimento de um aplicativo para auxiliar crianças autistas em seu cotidiano. O mesmo foi testado por uma equipe multidisciplinar, que deram um *feedback* positivo, no sentido que o aplicativo poderá contribuir na vida de uma criança autista de nível leve para moderado, melhorando suas relações sociais.

PALAVRAS-CHAVE Perturbações. Estudo. Inclusão. Multidisciplinar. Autista

RESUMO O autismo é um transtorno que era considerado raro, mas por meio de estudos epidemiológicos foi constatado que vem se tornando comum. Este transtorno apresenta perturbações que causam dificuldade nas relações sociais, problemas de comunicação e alterações bruscas de comportamento. As perturbações citadas excluem o indivíduo da sociedade, principalmente pelo não conhecimento do transtorno pela mesma, sendo que até profissionais e familiares encontram dificuldades na busca pela melhor forma de contribuir para o desenvolvimento do indivíduo com esse transtorno. Considerando

ABSTRACT Autism is a disorder that was considered rare, but epidemiological studies have shown that it is becoming common. This disorder presents disorders that cause difficulties in social relations, communication problems and abrupt changes in behavior. The mentioned disorders exclude the individual from the society, mainly due to the lack of knowledge of the disorder, and even professionals and family members find difficulties in the search for the best way to contribute to the development of the individual with this disorder. Considering these difficulties, a study was carried out to ascertain the existing technologies that aid in

the treatment of autism, so it was possible to observe the scarcity of quality software that take these premises into account. Therefore, a study was carried out to develop an application to help autistic children in their daily lives. It was tested by a multidisciplinary team, who gave positive feedback, in the sense that the application can contribute in the life of an autistic child from light to moderate level, improving their social relations.

KEYWORDS Disorders. Study. Inclusion. Multidisciplinary. Autistic

1 | INTRODUÇÃO

A sociedade, a partir do século XXI, tem buscado formas de quebrar o paradigma cartesiano, com padrões pré-estabelecidos, em que aqueles que não se adequam a ele, se sentem excluídos. A tendência é de fazer com que todas as pessoas, principalmente aquelas excluídas, tenham formas de se inserirem na sociedade como um todo e se sentirem aceitas. Muitas vezes, profissionais e a família ainda não sabem a melhor forma de colaborar com crianças que possuem necessidades específicas, em especial o autismo. Isto ocorre, principalmente, por essa síndrome ser mal compreendida, tendo apenas de 10 a 15% dos casos com causa genética específica. Mas essa síndrome desafia os pesquisadores que buscam, não somente tratamentos, mas soluções definitivas (GOMES, 2014). Diante de tal situação, torna-se importante conhecer as especificidades do transtorno e, assim, poder intervir, educacionalmente, de forma mais adequada.

O autismo é uma síndrome do comportamento humano, descoberta por KANNER (1997), no final da década de 1930; essa síndrome se caracterizou pela dificuldade das pessoas em manterem relações sociais, desde a infância, por deficiência de comunicação e linguagem, não conseguindo aceitar a mudança de rotina e no aprendizado. Apesar dessa síndrome não possuir causas definidas e nem cura definitiva ainda, ela pode ser diagnosticada precocemente e tratada para que as pessoas portadoras tenham uma melhor qualidade de vida. Há diferentes visões a respeito do autismo, mas o que se sabe é que não se separa o desenvolvimento cognitivo do desenvolvimento afetivo. Portanto torna-se fundamental conhecer as formas de abordagens de intervenções pedagógicas: ABA (Análise Aplicada do Comportamento), PECS (Sistema de Comunicação por meio de figuras), TEACCH (Programa de aprendizado individualizado), para serem aplicadas às crianças autistas. Essas intervenções serão abordadas na fundamentação teórica.

Por conseguinte, buscou-se alguma tecnologia de apoio, denominada tecnologia assistiva, que permitisse a utilização de intervenções pedagógicas para auxiliar no tratamento do autismo. Durante a busca foi possível observar que existem muitas tecnologias utilizadas para ensino de crianças, porém houve grande dificuldade em encontrar tecnologias desenvolvidas com foco em crianças autistas.

Em observância a essas premissas, foi levantada a hipótese da criação de um

aplicativo para dispositivo móvel que auxiliasse crianças autistas em suas rotinas diárias, de maneira interativa, com o auxílio dos pais, com o intuito de incluir essa criança na sociedade. Contudo, para criar um aplicativo para tratar essas crianças, de forma efetiva, faz-se necessário um estudo aprofundado sobre essa síndrome, as intervenções psicológicas e pedagógicas para tratamento, bem como técnicas de usabilidade, logo iniciou a definição objetivos, para o desenvolvimento do aplicativo.

O objetivo geral deste trabalho é desenvolver a proposta para criação do aplicativo para dispositivos móveis a ser aplicado no cotidiano de crianças com autismo, nos níveis leve e moderado. Os objetivos específicos são: 1. Estruturar o estudo, por meio de pesquisas bibliográficas. 2. Levantar dados, com profissionais que trabalham com crianças autistas: um educador físico, um fonoaudiólogo e um psicopedagogo. 3. Documentar os requisitos, utilizando técnicas de engenharia de software. 4. Analisar os documentos, levantando possibilidades de adequação ao processo de autismo. E, por fim, 5. Criar o protótipo do aplicativo, com base na documentação levantada, por meio das fases anteriores.

A presente pesquisa se justifica pelo desejo de utilizar a tecnologia para dar mais autonomia e inclusão social para pessoas que tenham algum tipo de deficiência. Neste contexto, foram pesquisados vários tipos de deficiências: Físicas, Visuais e Neurológicas, bem como os recursos tecnológicos que já existem com intuito de auxiliar as pessoas supracitadas. Por meio das informações levantadas na internet, pôde-se notar que existe um número maior de intervenções, utilizando tecnologias para pessoas com deficiência física, do que para pessoas com deficiências neurológicas. Esse cenário encontrado contribuiu para que fosse definido que o trabalho iria vislumbrar o tratamento de uma deficiência neurológica. Por conseguinte, foi realizada uma investigação sobre algumas deficiências neurológicas, onde uma delas chamou a atenção, o autismo, por estar associada a crianças e apresentar uma evolução significativa do número de casos dessa patologia. Estes fatos motivaram o desenvolvimento de um aplicativo móvel para tratamento do autismo.

Nos próximos capítulos serão apresentados: A fundamentação teórica sobre o autismo; as tecnologias que serviram de base para o desenvolvimento do aplicativo e as etapas para o desenvolvimento do aplicativo, desde sua concepção até o seu fechamento. Por fim foram apresentadas as considerações finais do trabalho.

2 | FUNDAMENTAÇÃO TEÓRICA

O objetivo deste capítulo é fornecer o embasamento teórico e a contextualização do autismo, intervenções educacionais, tecnologias para o tratamento dessa patologia e as técnicas e tecnologias utilizadas para o desenvolvimento deste trabalho.

2.1 Autismo

A primeira publicação sobre autismo foi feita por Leo Kanner, médico austríaco, residente em Baltimore nos EUA (Estados Unidos da América). Em 1943, ele escreveu o histórico artigo: “Os distúrbios autísticos de contato afetivo”. Nesse artigo, Kanner descreveu o estudo feito com crianças em idade inferior a 11 anos, que apresentavam comportamentos diferentes e singulares de outras crianças observadas, até então. Foram descritos 11 casos (oito meninos e três meninas), sendo o primeiro Donald T., avaliado pela primeira vez em outubro de 1938 e a última Elaine C., levada pelos pais em 12 de abril de 1939. Foi observado nessa pesquisa que, apesar das diferenças individuais, existiam características comuns, essenciais e inevitáveis. A principal característica constatada foi que essas crianças tinham uma incapacidade no relacionamento interpessoal, que as distinguia de outras patologias como a esquizofrenia: “o distúrbio fundamental mais surpreendente é a incapacidade dessas crianças de estabelecer relações de maneira normal com as pessoas e situações, desde o princípio de suas vidas”. (KANNER, 1997).

Em 1944, o também austríaco Hans Asperger, com formação na Universidade de Viena, na Itália, escreveu um artigo em alemão: “Psicopatologia autística da infância”, que descreveu características apresentadas por crianças, bem semelhantes às descritas por Kanner. Porém, esse artigo demorou muito tempo para se tornar conhecido da população científica internacional, permaneceu desconhecida até Uta Frith traduzi-lo para Inglês. (ASPERGER, 1994).

Esses autores foram os precursores da identificação do autismo e, posteriormente, foram estudados por outros pesquisadores. BOSA (2000) define a criança autista, como aquela que tem inaptidão para estabelecer relações normais com o outro, uma demora na aquisição da fala e, no momento em que ela se desenvolve, uma incapacitação de lhe dar um valor de comunicação. Como é muito difícil imaginar todos estes desvios juntos, a pesquisadora Francesca Happé, propõe, em seu livro, que as pessoas ao tentarem imaginar-se “em um país com cultura e língua desconhecidas, com as mãos imobilizadas, sem compreender os outros e sem possibilidade de se fazer entender”. (HAPPÉ & BERNARDOS, 2007).

O autismo, há décadas, foi considerado uma condição relativamente rara, mas, dados epidemiológicos têm alterado radicalmente essa percepção. Com base em uma ampla pesquisa realizada pelo CDC (Centro de Controle e prevenção de doenças), nos EUA, estimou a ocorrência de autismo em 1 a cada 88 crianças, com maior frequência (5 vezes) em crianças do sexo masculino: 1 a cada 54. E, no sexo feminino, 1 a cada 252. No entanto, estudos epidemiológicos são difíceis de comparar, pois há diferenças na população pesquisada, nos mecanismos de recrutamento, tamanho da amostra, critérios de diagnósticos, instrumentos utilizados, bem como se critérios de imparidade são incluídos (FOMBONNE, 2009). Mesmo assim, utilizando a mesma metodologia

ao longo de um período de oito anos, foi criada uma rede de monitoramento que encontrou aumento nas taxas de autismo nos EUA, conforme apresentado na **Tabela 1**.

Identified Prevalence of Autism Spectrum Disorder ADDM Network 2000-2010 Combining Data from All Sites				
Surveillance Year	Birth Year	Number of ADDM Sites Reporting	Prevalence per 1,000 Children (Range)	This is about 1 in X children...
2000	1992	6	6.7 (4.5 – 9.9)	1 in 150
2002	1994	14	6.6 (3.3 – 10.6)	1 in 150
2004	1996	8	8.0 (4.6 – 9.8)	1 in 125
2006	1998	11	9.0 (4.2 – 12.1)	1 in 110
2008	2000	14	11.3 (4.8 – 21.2)	1 in 88
2010	2002	11	14.7 (5.7 – 21.9)	1 in 68

Tabela 1 – Tabela de Identificação de prevalência de transtornos do espectro do autismo
Fonte: CDC,2016

São várias as possíveis causas do autismo, porém as mais aceitas estão ligadas a neurociência, que pesquisa sobre anormalidades em alguma parte do cérebro humano, porém ainda não se conseguiu definir de forma conclusiva. Segundo MOURA et al (2005):

Por não existir marcadores biológicos, o diagnóstico do autista é ainda baseado em critérios comportamentais, mesmo que as pesquisas sobre o tema estejam em constante avanço, ainda há uma distância muito grande na compreensão desse fenômeno (MOURA et al, 2005, p. 112).

Mesmo assim, estas evidências apontam para a melhor forma de diagnosticar o autismo, que pode se manifestar precocemente por meio de perturbações, geralmente antes dos três anos de idade.

2.2 Perturbações

WING (1979) separa as perturbações que as crianças autistas possuem em três diferentes áreas de domínio: 1. Social; 2. Linguagem e Comunicação; 3. Comportamento e Pensamento. Todavia BAPTISTA & BOSA salientam que, apesar de Lorna Wing separar as três perturbações (triade), elas não são separáveis, como leva a crer o termo “triade”, “a expressão resultou de mensurações estatísticas,

demonstrando que os comprometimentos que apareciam nessas áreas não ocorriam ao acaso; apresentavam-se juntos, embora com intensidade e qualidades variadas” (BAPTISTA & BOSA, 2002, p.34).

Na área social, o autista não consegue interagir para compreender as regras sociais, por isso tem muita dificuldade em se relacionar com outros indivíduos. O autista evita o contato visual, não possui expressões faciais e necessita ter uma rotina; as crianças podem tanto se isolar quanto interagir de forma estranha aos padrões habituais. Essa não interação fez Kanner afirmar: “há nelas a necessidade poderosa de não serem perturbadas. Tudo o que é trazido para a criança do exterior, tudo o que altera o seu meio externo ou interno representa uma intrusão assustadora” (KANNER, 1997). Porém TREVARTHEN, (1996) confronta Kanner por meio de estudos e diz: “nem todos os autistas mostram aversão ao toque ou isolamento, alguns, ao contrário, podem buscar o contato físico, inclusive de uma forma intensa, quando não ‘pegajosa’, segundo pais e professores”.

Logo, pode-se notar que essa área está relacionada à dificuldade do autista de assimilar o sentimento, o pensamento e a reação das pessoas, pois para os autistas é de extrema complexidade, expressar os seus sentimentos.

A área da linguagem e comunicação mostra que o autista não consegue iniciar e nem manter uma conversa, seja utilizando uma linguagem verbal ou não verbal e, na ânsia de tentar se comunicar, ele utiliza uma linguagem estereotipada e repetitiva. Estudando esse comportamento, LAMÔNICA (1992) constatou que três quartos das crianças autistas que conseguiram falar, possuíam ecolalia. Ela conseguiu também estimar, que metade das crianças com autismo, não conseguem desenvolver a linguagem e que é raro encontrar autistas que falam normalmente.

Na área do comportamento e pensamento, pode-se notar que o autista tem uma imaginação limitada, devido a uma inflexibilidade de pensamentos e comportamentos, ele tem comportamento obsessivo e é ritualista. A criança autista não identifica um objeto por inteiro, ela consegue identificar apenas uma parte ou detalhe, com isso fica difícil entender a função desse objeto que pode ser um brinquedo, isso acarreta um atraso intelectual, por isso se faz necessário o cumprimento de rotinas, que auxiliam na assimilação dos objetos.

2.3 Níveis de autismo

Os níveis de autismo são definidos pela DSM (Diagnostic and Statistical Manual of Mental Disorders), que é um guia publicado pela Associação Psiquiátrica Americana, no qual os médicos se baseiam para fornecer um diagnóstico formal.

A DSM se encontra na quinta edição DSM-5, que é uma tentativa de dar aos médicos uma lista com todas as condições reconhecidas de saúde mental, incluindo os seus sintomas, por isso tem grande influência na CID (Classificação Internacional de Doenças), elaborada pela OMS (Organização Mundial da Saúde), que trata de

todas as doenças reconhecidas. Essa influência pode ser notada, pois a OMS estuda as atualizações da 10ª Revisão (CID-10) para a 11ª Revisão (CID-11), principalmente no que diz respeito à saúde mental. (FRAZIER, 2012). A principal mudança da DSM-5, em relação à edição anterior, é que foi criado o termo TEA (Transtorno do Espectro do Autismo), que engloba todos os tipos de sintomas autísticos. E, para diferenciar esses sintomas, foram criados três níveis: leve, moderado e severo, que são analisados por meio da gravidade, baseados na necessidade do apoio destinado, devido aos desafios com a comunicação social, interesses restritos e comportamentos repetitivos.

O nível leve ou grau leve se apresenta nos indivíduos com TEA, que manifestam apenas alguns sintomas, como a dificuldade na interação social, onde as habilidades cognitivas são preservadas; esses indivíduos necessitam ter acompanhamento apenas na infância (EDGIN & PENNINGTON, 2005). O nível moderado ou grau moderado (clássico) é constituído por indivíduos com TEA que falam, mas não se comunicam, são capazes de reportar frases fora de contexto, mas não conseguem responder uma pergunta. Aprendem apenas o sentido literal das palavras, não compreendem metáforas nem duplo sentido e têm ligações muito pobres com o ambiente e precisam de acompanhante até se tornar jovens (SCHWARTZMAN, 2011).

No nível severo ou grau severo, os indivíduos com TEA, apresentam deficiência intelectual considerada grave, vivendo de forma isolada, pois não conseguem se relacionar, visto que sua linguagem verbal e não verbal é praticamente ausente. Esses indivíduos não têm discernimento, malícia, não manifestam afeto, eles têm aversão à mudança de rotina, e quando acontece uma mudança, se descontrolam totalmente, podendo até provocar crises de agressividade. Por isso, eles precisam de acompanhamento até para realizar atividades básicas, como se vestir, ir ao banheiro, entre outros, sendo totalmente dependentes de um acompanhante (VINOCUR, 2014).

A pessoas com TEA, manifestam uma vasta gama de severidade e prejuízos, sendo frequentemente a causa de deficiência grave, representando um grande problema de saúde pública. Há uma grande heterogeneidade na apresentação fenotípica do TEA, tanto com relação à configuração e severidade dos sintomas comportamentais (GESCHWIND, 2009).

O autismo é um distúrbio, ainda, incurável, mesmo nos níveis mais leves, porém se a criança for diagnosticada nos primeiros anos de vida e houver um estímulo por parte da família, poderá obter ganhos significativos, principalmente nos casos de menor deficiência mental (SCHWARTZMAN, 2011).

2.4 A Família do Autista

A família exerce um fundamental papel na socialização de uma pessoa, ela é a matriz da aquisição de conhecimento humano e precursora das relações sociais, afetivas e cognitivas de um indivíduo. Segundo DESSEN & BRAZ (2005), é na experiência familiar que ocorre a formação de conjuntos comportamentais, de ações e resoluções

de problemas. Sendo assim, os principais estudos sobre a família e o autismo, visam apontar quais são as influências familiares que agem negativamente na afetividade, que pudesse promover a incapacidade relacional do autista. Com esses estudos, foi possível observar que a maioria dos genitores eram dotados de intelectualidade e eram afetivamente frios. SPROVIERI & ASSUMPTÃO JR (2001), por meio de uma pesquisa, discutiram que “a característica do autismo expõe os pais a um luto permanente da criança saudável que acompanha todas as fases do desenvolvimento”. Por isso, o diagnóstico de autismo deve ser discutido com a família para evitar expectativas irreais e, para que a família possa concentrar todos os esforços em uma intervenção precoce, buscando maior conhecimento e envolvimento. Com maior conhecimento, os pais poderão identificar o que se deve ensinar a uma criança autista e terão condições de encontrar estratégias para remediar o atraso do desenvolvimento social do autista, isso favorecerá o desenvolvimento da criança diagnosticada.

2.5 Intervenções educacionais

Esse tópico irá abordar os principais tipos de intervenções educacionais (métodos de aprendizagem) que auxiliam os pais e os educadores, a desenvolverem atividades com crianças autistas.

A ABA (Análise Aplicada ao Comportamento) é um tratamento comportamental, que objetiva ensinar habilidades que a criança não as detém, por meio de fases bem definidas, pois o autista tem grande dificuldade em generalizar estímulos. Essas habilidades são ensinadas, geralmente, de maneira individualizada, onde a criança autista aprende por meio de instruções e, quando ela consegue realizar determinada atividade, ela recebe um estímulo, o que torna a atividade prazerosa. Inicialmente, são ensinadas habilidades básicas como: realizar o contato visual, sentar, esperar sua vez, para depois partir para as situações de grupo e pedagógicas. (MELLO, 2007).

O PECS (Sistema de Comunicação por troca de Figuras) é um método considerado simples e de baixo custo, que tem o objetivo de organizar a comunicação por meio de cartões, que funcionam, tanto para linguagens verbais quanto para as não-verbais. Esse método, apesar de simples, quando bem implementado, produz resultados inquestionáveis, levando o autista a conseguir expressar de forma espontânea, seus desejos, além de melhorar a interação social. (SAMPAIO, 2005).

O último método a ser exposto é o TEACCH (Tratamento e Educação para Crianças Autistas e com Distúrbios na Comunicação), que é um dos mais utilizados, por fazer uma organização do ambiente físico, deixando a criança autista mais segura e confortável. Esse método foi proposto, pois foi observado que quando um autista é levado para um ambiente desconhecido, causa inquietação, irritação e até agressividade. Isso tudo acontece quando a situação foge do habitual (FONSECA, 2016).

Esses métodos não são excludentes e sim complementares e auxiliam o autista a adquirir habilidade por meio de rotinas na sua vida diária. Segundo NILSSON (2004,

p.52-53), “O autista apresenta um pensamento literal concreto, visual, fragmentado”, isso mostra a importância da utilização dos métodos apresentados, visto que todos têm natureza visual. Portanto, oferecer uma ferramenta visual e rotineira a uma criança com autismo, além de apresentar as informações de maneira mais compreensíveis, irá estimular o desenvolvimento da aprendizagem.

2.6 Tecnologias Envolvidas

Para se definir as tecnologias a serem aplicadas ao projeto, utilizou-se os conceitos de Engenharia de Software, que é a área do conhecimento da engenharia que abrange todos os aspectos da produção de software, desde os estágios iniciais de especificação do sistema até a manutenção do mesmo. Com isso buscou-se uma tecnologia de apoio, denominada de tecnologia assistiva, que ajuda um indivíduo com deficiência, em suas atividades diárias. Há relatos de pessoas com deficiência que utilizaram essa tecnologia, se tornaram e permaneceram independentes. (COOK & POLGAR, 2015).

É notável que os estudos sobre essa tecnologia, no Brasil, estão progredindo a cada ano, e por isso, é importante assegurar políticas públicas que estabeleçam e incentivem o uso da tecnologia assistiva, com o objetivo de contribuir para o desenvolvimento e utilização dessas tecnologias para as pessoas com necessidades específicas. O Ministério da Educação desenvolve políticas de educação inclusiva, como o AEE (Atendimento Educacional Especializado, que “é um serviço da educação especial que identifica, elabora, e organiza recursos pedagógicos e de acessibilidade, que eliminem as barreiras para a plena participação dos alunos, considerando suas necessidades específicas” (SEESP/SEED/MEC, 2007).

Nessa perspectiva, há uma necessidade em fundir os conceitos de tecnologias assistivas e atendimento educacional especializado, para se criar uma educação inclusiva, onde o portador de deficiência poderá ter a possibilidade de desenvolver suas atividades cotidianas, colocando-o em situações semelhantes às das pessoas que não necessitam de atendimento especializado. Para essa educação se tornar possível, há necessidade da utilização de recursos (equipamentos) que variam entre uma simples bengala a um sistema computadorizado complexo e completo (BERSCH, 2006).

Dentre os equipamentos de tecnologia assistiva, uma que se destaca é a utilização de dispositivos móveis, por serem tecnologias com muita disponibilidade no mercado mundial.

O protótipo foi desenvolvido levando em consideração conceitos de usabilidade.

A usabilidade é um atributo de qualidade e refere-se à facilidade com que o usuário interage com a interface, ou seja, a rapidez na aprendizagem, a eficiência com que este utiliza os recursos disponíveis para executar uma determinada tarefa, o quanto lembram de como usar um recurso já conhecido, a propensão à erros e a motivação a usar os recursos disponíveis novamente, concluindo que se alguns

recursos não puderem ou não forem utilizados podem muito bem não existirem. (NIELSEN e LORANGER, 2007, p. 16)

Ausabilidade está inserida dentro da área chamada interação homem-computador e é multidisciplinar, pois suas avaliações e métodos são apoiados por diversas disciplinas como: Psicologia, Economia, Ciência da Computação, dentre outros.

Para documentar o aplicativo foi utilizado os diagramas de Casos de Uso e de Atividades da linguagem Unified Modeling Language – UML, que é uma linguagem visual de modelagem de programas com o paradigma de programação orientado a objetos. Ela fornece uma boa apresentação para o usuário de como será desenvolvido o software por meio de detalhes estruturais minuciosos e aspectos comportamentais de um único sistema, fornecendo vários pontos de vistas, para que o usuário possa ter uma ideia de como funcionará o software antes de sua implementação (LARMAN, 2007).

3 | METODOLOGIA

As abordagens modernas de sistemas adotam a filosofia baseada em modelos. O que varia bastante entre os métodos de um sistema, são os tipos de modelos que devem ser construídos, como devem ser construídos e quem deve construí-los. Por esse motivo, com a utilização dos conceitos de Engenharia de Software para produção do aplicativo, foi escolhido o Modelo Cascata, demonstrado na **Figura 1**, por ser um método que deixa cada etapa do desenvolvimento bem definidas, o que contribui para um ganho substancial de tempo. Neste modelo, consideram-se as atividades de especificação, desenvolvimento, validação e evolução, fundamentais ao processo e as representam como fases separadas do mesmo: a especificação de requisitos, o projeto de software, a implementação, os testes e, assim por diante. Seu nome é devido à sequência em cascata de uma fase para outra (SOMMERVILLE, 2004).

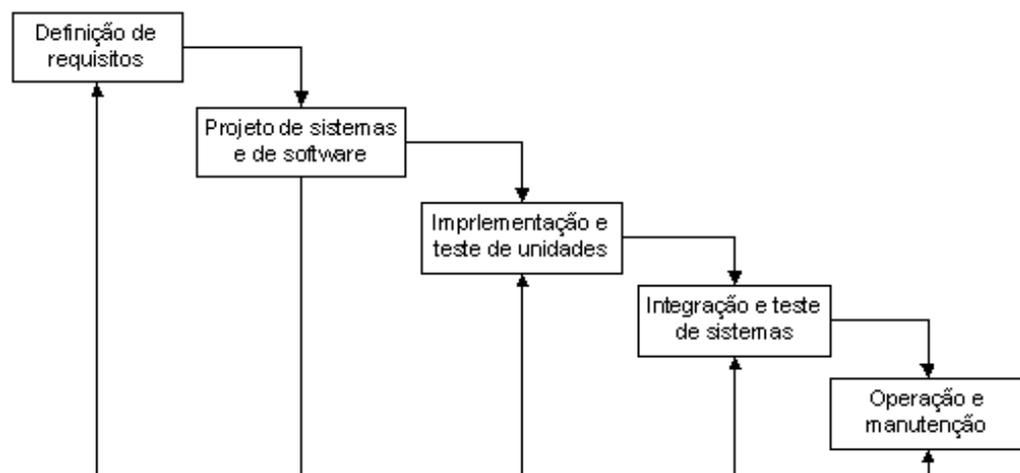


Figura 1: Modelo Cascata

Fonte: SOMMERVILLE, 2004, p.38

Segundo a metodologia de desenvolvimento em cascata, inicialmente, foi

realizada a Definição dos requisitos, que consistiu em uma pesquisa qualitativa descritiva e exploratória utilizando métodos de pesquisa científica tecnológica, que iniciou com a busca por cursos sobre autismo e posteriormente a busca de profissionais que trabalham no tratamento do autismo. Esses profissionais foram encontrados e logo foi montado uma equipe multidisciplinar, que contribuíram para a pesquisa por meio de entrevistas e reuniões, que resultaram na definição de um aplicativo para crianças autistas de níveis leve e moderado, que tem como objetivo auxiliar os pais dessas crianças a regularem suas atividades diárias, utilizando como intervenções pedagógicas, a agenda do TEACCH e o conceito de cartões do PEC. Com isso foram levantados e definidos os principais requisitos que o aplicativo iria contemplar, logo foi feita a elaboração da documentação do sistema levando em consideração toda a análise realizada na etapa anterior. Esta documentação foi modelada por meio dos diagramas da UML, e apresentada a equipe multidisciplinar, para que esses diagramas pudessem passar por reajustes antes do início do desenvolvimento, a **Figura 2** apresenta o diagrama de Caso de Uso do Aplicativo.



Figura 2: Diagrama de Casos de Uso do Aplicativo

Fonte: Dados do próprio trabalho

Na etapa, Projeto de sistema de software, foram definidas a estruturação dos dados, a arquitetura do software e a caracterização da interface. Com as etapas anteriores completadas, passou-se para a próxima etapa: Implementação e testes de unidade, onde foi desenvolvido um protótipo do aplicativo, que é uma representação

do sistema com algumas funcionalidades básicas; este desenvolvimento foi realizado levando em consideração os conceitos de usabilidade, o protótipo passará por simples testes sistêmicos e adequações sugeridas pela equipe como: colocação de limites visuais, colocação de áudio motivacional a cada acerto, não permitir arrastar a sequência errada e a disposição dos componentes na tela, para a utilização ser mais sugestiva. Chegou-se então a uma versão de apresentação observada na **Figura 3**, que foi utilizada pela equipe, onde as envolvidas analisaram a efetividade do protótipo, e deram um *feedback*, dizendo que o aplicativo superou suas expectativas, mostrando-se um aplicativo intuitivo, que seguiu todas as orientações passadas, e que o mesmo poderá ser colocado em testes com crianças autistas de nível leve e moderado. Este *feedback*, foi conseguido, após ser disponibilizado o aplicativo para a equipe testar durante uma semana, ao final dessa semana foi realizada uma reunião, onde as mesmas mostraram seus pontos de vista.

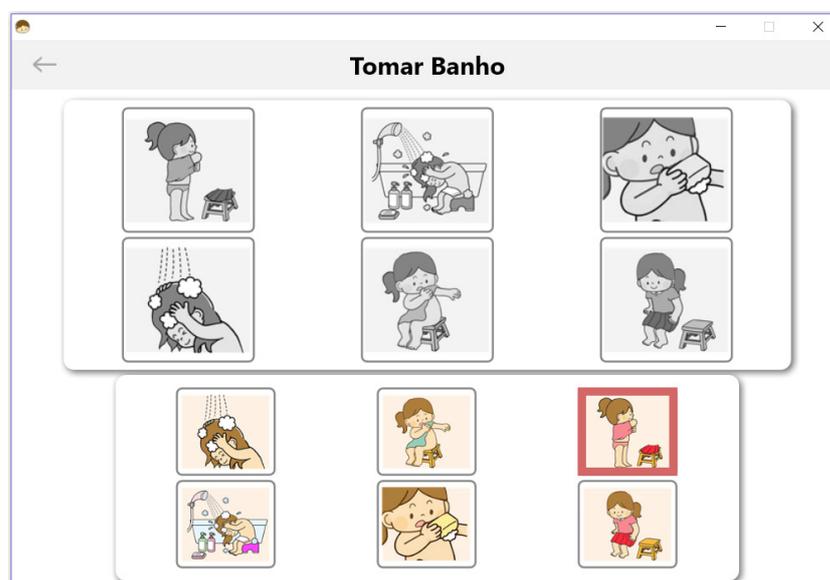


Figura 3: Exemplo de uma das telas do aplicativo
Fonte: Dados do próprio trabalho.

4 | CONSIDERAÇÕES FINAIS

Até onde a pesquisa foi realizada, notou-se uma necessidade enorme de aplicativos direcionados a exercícios para crianças autistas. Por sua demonstração das atividades e ferramentas aplicadas a seus clientes, percebe-se que, realmente, há necessidade de maiores pesquisas na área de aplicativos em dispositivos móveis (principalmente, por ser mais cômodo para a criança e para a família) e outras ferramentas que possam colaborar para o desenvolvimento da criança para promover a inclusão. Pela pesquisa bibliográfica, pode ver que é um grande desafio para a família e para os profissionais, uma vez que a perturbação é bastante complexa, exigindo dos profissionais muita pesquisa e criação de ferramentas adequadas ao autismo.

A pesquisa está com o protótipo pronto com muitas funcionalidades, porém

ainda necessita de ser testada com crianças autistas e passar por ajustes para se ter uma versão final, para ser colocado à disposição para download. Mas o protótipo foi testado e muito aceito pela equipe multidisciplinar que contribuiu para a elaboração do aplicativo, onde a equipe se surpreendeu com os detalhes das intervenções pedagógicas TEACCH e PEC apresentadas no protótipo do aplicativo, deixando um feedback satisfatório sobre a efetividade que poderá ter o aplicativo se utilizado por crianças autistas

Espera-se que, ao ser utilizado por crianças autistas, o aplicativo possa contribuir significativamente para o tratamento das mesmas.

REFERÊNCIAS

ASPERGER, A. Autistic psychopathy in childhood. In U. Firth (Ed), **Autism and Asperger syndrome** (pp.37-92). Cambridge: Cambridge University Press, 1994.

BAPTISTA, Cláudio Roberto; BOSA Cleonice; e colaboradores. **Autismo e educação: reflexões e propostas de intervenção**. Porto Alegre, Artmed, 2002.

BERSCH, R. Tecnologia assistiva e educação inclusiva. In: **Ensaio Pedagógico**. Brasília: SEESP/MEC, 2006, p. 89-94.

BOSA, Cleonice; CALLIAS, Maria. **Autismo**: breve revisão de diferentes abordagens. *Psicol. Reflex. Crit.* V. 13 n. 1 Porto Alegre, 2000. Disponível em: <http://www.scielo.br/scielo.php?pid=S010279722000000100017&script=sci_abstract&lng=p> Acesso: 9 nov 2015.

CDC. **Centers for Disease Control and Prevention**. Disponível em:< <http://www.cdc.gov/ncbddd/autism/data.html> > Acesso: 29 fev 2016.

COOK, A. M.; POLGAR, J. M. Cook & Hussey's **Assistive Technologies**: Principles and Practice. Third Edition, Mosby Elsevier, Philadelphia, USA, 2008.

DESSEN, M. A., & BRAZ, M.P.A família e suas inter-relações com o desenvolvimento humano. In: M.A Dessen & Á.L. Costa Jr. (Orgs.). **A ciência do desenvolvimento humano: tendências atuais e perspectivas futuras** (pp.113-131). Porto Alegre: Artmed,2005.

DSM5. **American Psychiatric Association**. Disponível em: <<http://dsm5.org/>> Acesso: 18 nov 2015.

EDGIN J.O, PENNINGTON B.F. **Spatial cognition in autism spectrum disorders**: Superior, impaired, or just intact? *J Autism Dev Disord* 2005.

FOMBONNE E. **Epidemiology of pervasive developmental disorders**. *Pediatric Research*, 65:591-598.2009.

FONSECA, Maria Elisa Granchi; CIOTA, Juliana De Cássia Baptistella. **Vejo e Aprendo - Fundamentos do Programa Teacch - o Ensino Estruturado Para Pessoas Com Autismo**. Ribeirão Preto: Booktoy. 136p, 2016.

FRAZIER T, YOUNGSTROM E, SPEER L. Validation of proposed DSM-5 criteria for autism spectrum disorder. **Journal of the American Academy of Child and Adolescent Psychiatry**, 51:28-40.2012.

GESCHWIND, D.H. Advances in Autism. *Annual Review of Medicine*, v. 60, p. 367-380, 2009.

GOMES, Marina. Biologia do Autismo. **Ciência e Cultura**. Vol. 66 n. 1. São Paulo, 2014. Disponível em:< http://cienciaecultura.bvs.br/scielo.php?pid=S0009-67252014000100004&script=sci_arttext>

Acesso: 5 de maio 2016.

HAPPÉ, Francesca, BERNARDOS, Maria N. *Introducción al autismo*. Malaga: Agapea, 2007.

KANNER, L. Os distúrbios autísticos de contato afetivo. In: ROCHA, P. S. **Autismos**. São Paulo: Escuta, 1997.

LAMÔNICA, Dionísia Aparecida Cusin. **Utilização de variações da técnica do ensino incidental para promover o desenvolvimento da comunicação oral de uma criança diagnosticada autista**. Bauru, USC, 1992. (Cadernos de divulgação cultural)

LARMAN, C. **Utilizando UML e padrões**: uma introdução à análise e ao projeto orientados a objetos e ao desenvolvimento iterativo. 1. ed. São Paulo: Bookman, 2007.

MELLO, Ana Maria S. Rios. **Autismo: guia prático**. 7ª ed. São Paulo: AMA; Brasília: Corde, 2007

MOURA, Paula J.; SATO, Fabio; MERCADANTE, Marcos T. **Bases Neurobiológicas do Autismo**: Enfoque no domínio da sociabilidade in **Caderno de Pós-graduação em Distúrbios do Desenvolvimento**. 2005. Disponível em: <http://www.mackenzie.br/fileadmin/Pos_Graduacao/Mestrado/Disturbios_do_Desenvolvimento/Publicacoes/volume_IV/000.pdf> Acesso: 8 nov 2016.

NIELSEN, J.; LORANGER, H. **Usabilidade na Web**: Projetando Websites com qualidade. 5. ed. Rio da Janeiro: Elsevier, 2007.

NILSSON, Inger. Introdução a educação especial para pessoas com transtornos de espectro autístico e dificuldades semelhantes de aprendizagem. Em PDF. **Congresso Nacional sobre a Síndrome de Autismo 2004**. Disponível em < <http://www.ama.org.br/download/Autismo-IntrodEducEspecial.pdf>> Acesso: 23 mar. 2016.

SAMPAIO, A. S. **Transtorno autista e a abordagem cognitivo-comportamental**: possibilidade de auxílio psicológico. 2005. Disponível em: www.psicologiavirtual.com.br. Acesso: 29 de out 2015.

SCHWARTZMAN, J.S. **Em entrevista ao Drauzio Varella**. Publicado em: 12 dezembro 2011. Disponível em: <<http://drauziovarella.com.br/crianca-2/autismo-primeira-parte/>> Acesso: 25 de jan 2016.

SEESP/SEED/MEC (2007). **Atendimento educacional especializado**. Disponível em: <http://portal.mec.gov.br/seesp/arquivos/pdf/aee_df.pdf> Acesso: 15 de jan 2016.

SOMMERVILLE, Ian. **Engenharia de Software**. 6.ed. São Paulo: Pearson Education, 2004.

SPROVIERI, M.H, & ASSUMPÇÃO JR., F.B. **Dinâmica familiar de crianças autistas**. Arquivos Neuropsiquiatria, 59(2-A), 230-237.2001.

TREVARTHEN, C., AITKEN, K., PAPOUDI, D. & ROBARTS, J. **Children with autism**: Diagnosis and interventions to meet their needs. London: Jessica Kingsley, 1996.

VINOCUR, N. (2014). **May 22. France's Sarkozy calls for two-speed EU, tighter borders**. Disponível em:<<http://www.reuters.com/article/2014/05/22/us-france-sarkozy-idUSBREA4L07120140522>> Acesso em:18 jun. 2016.

WING, L. GOULD, J. (1979), Severe Impairments of Social Interaction and Associated Abnormalities in Children: **Epidemiology and Classification**, Journal of Autism and Developmental Disorders, 9, pp. 11–29

APRENDIZAGEM SIMULADA NA NUVEM

Rafaela R. Jardim

Universidade Federal do Rio Grande do Sul
(UFRGS)
Porto Alegre – RS

Roseclea Duarte Medina

Universidade Federal de Santa Maria (UFSM)
Santa Maria – RS

Giliane Bernardi

Universidade Federal de Santa Maria (UFSM)
Santa Maria – RS

Fabrizio Herpich

Universidade Federal do Rio Grande do Sul
(UFRGS)
Porto Alegre – RS

Andressa Facalde

Universidade Federal de Santa Maria (UFSM)
Santa Maria – RS

Eduardo Lemos

Instituto Federal Farroupilha Campus São
Vicente do Sul (IFFar)
São Vicente do Sul – RS

RESUMO As soluções de Cloud Computing podem ser utilizadas para apoiar a aprendizagem em ambientes U-learning e são uma alternativa atrativa para maximizar/flexibilizar os recursos computacionais. Dessa forma, este trabalho objetiva a apresentação do laboratório virtual U-Lab Cloud, ao qual foi associado à tecnologia de Cloud Computing e as características de U-learning. Foi realizado um estudo de caso

simulado, onde foi possível verificar a evolução dos estilos de aprendizagem de estudantes fictícios em um processo educacional. Com os resultados obtidos, foi possível demonstrar a viabilidade da adoção da tecnologia de Cloud Computing em ambientes educacionais.

PALAVRAS-CHAVE Nuvem; Aprendizagem; U-Learning;

ABSTRACT Cloud Computing solutions can be used to support learning in U-learning environments and are an attractive alternative to maximize / flexible computing resources. Thus, this study aims to present a ubiquitous virtual laboratory U-Lab Cloud, which was associated with the technology of Cloud Computing and the U-learning features. a simulated case study, it was possible to check the evolution of the learning styles of students in a fictional educational process was carried out. With the results obtained, it was possible to demonstrate the feasibility of adopting cloud computing technology in educational settings.

KEYWORD Cloud; Learning; U-Learning;

1 | INTRODUÇÃO

A evolução das Redes de Computadores influenciou tanto em tamanho como em complexidade de infraestrutura. Para Filippetti

(2008), esses avanços fizeram crescer a demanda e a exigência por profissionais melhor qualificados, capazes de suportar todas as nuances tecnológicas desses ambientes. Com isso, evidencia-se a importância da formação de profissionais aptos a planejar e desenvolver atividades técnicas na área de Redes de Computadores. Dessa forma, as aulas práticas possibilitam a experimentação dos conceitos vistos pelos estudantes. Dado tal cenário, torna-se necessário buscar novas formas e estratégias para o ensino de Redes de Computadores contemplando a utilização dos recursos tecnológicos disponíveis atualmente.

Diante de diversas ferramentas que apresentam grande potencial educacional, tem-se os ambientes virtuais imersivos ou também chamados mundos virtuais, que permitem experiências de simulação e atividades de colaboração. O paradigma educacional imersivo tem como objetivo disponibilizar espaços tridimensionais onde o estudante pode transitar e vivenciar experiências em um ambiente altamente interativo (ORGAZ et al. 2012). Neste cenário, surge a necessidade de propiciar ambientes virtuais que forneçam recursos computacionais adaptados às particularidades de cada estudante.

Para a adaptação e personalização de ambientes educacionais podem ser utilizados elementos de ubiquitous learning (*U-Learning*). Yahya et al. (2010) definem *U-Learning* como um paradigma de aprendizagem localizado em ambientes computacionais ubíquos, que permite o estudante aprender, no lugar, no tempo e na direção correta. Existem diversos aspectos que devem ser introduzidos na projeção dos ambientes *U-Learning*, como as informações de contexto.

Conforme Dey (2001), o contexto é qualquer informação que possa ser usada para caracterizar a situação de entidades que sejam consideradas relevantes para a interação entre um usuário e uma aplicação. Herpich et al. (2016) corroboram afirmando que, a partir das informações contextuais coletadas, e.g. contexto computacional, físico e de tempo, diversas são as possibilidades de utilização das mesmas em aplicações computacionais, destacando-se por exemplo, o fornecimento de serviços personalizados aos usuários, tais como a adaptação de conteúdos e ferramentas conforme as preferências dos mesmos.

A tecnologia de *Cloud Computing* (Computação em Nuvem), quando utilizada no campo educacional, pode impactar positivamente na construção, manutenção e atualização de infraestruturas. As soluções de *Cloud Computing* podem ser utilizadas para apoiar a aprendizagem em ambientes *U-learning*, pois são alternativas para maximizar e flexibilizar os recursos computacionais.

Na presente pesquisa foi implementado um ambiente de nuvem privada para hospedar o laboratório virtual ubíquo U-Lab Cloud. Esse ambiente apresenta características de um sistema *U-learning*, sendo adaptável ao contexto dos estudantes e foi disponibilizado por meio da tecnologia de *Cloud Computing*. Desse modo, as informações de contexto consideradas nesse trabalho são: o estilo cognitivo, o tipo de dispositivo e a velocidade de rede.

Este artigo está organizado da seguinte forma: na seção II são explanados os conceitos relacionados a *Cloud Computing* e a *U-learning*; a seção III descreve a metodologia utilizada nesta pesquisa; a seção IV apresenta o ambiente desenvolvido; na seção V encontra-se a implementação realizada; já a análise dos resultados encontra-se na seção VI; por fim, na seção VII estão as considerações finais deste trabalho.

2 | REFERENCIAL TEÓRICO

A fim de contribuir nas atividades realizadas durante esta pesquisa, foi desenvolvida uma revisão bibliográfica sobre os temas relevantes para o estudo. As seções a seguir apresentam uma visão geral dos conceitos sobre *Cloud Computing* e *U-learning*.

2.1 Cloud Computing

Na literatura encontram-se diversas definições para *Cloud Computing* como pode ser visto em (Caminero et al. 2013; Wang et al. 2010), sendo a mais apropriada para este trabalho a definição cunhada por Mell e Grance (2011), onde *Cloud Computing* é definida como um modelo computacional que permite o acesso sob demanda da rede a um pool de recursos computacionais configuráveis que podem ser liberados com um esforço mínimo de gerenciamento.

A utilização da *Cloud Computing* pode auxiliar os usuários a se libertarem das amarras da tecnologia e contribuir para a construção da sua aprendizagem (BAI, 2011), uma vez que toda a aplicação encontra-se disponível na “nuvem”. Assim, para usufruir da sua aplicação, os usuários precisam apenas de um navegador simples com conectividade à Internet (JADEJA e MODI, 2012). Além disso, outra vantagem para o usuário desse tipo de aplicação é que a mesma dispensa atualização de *hardware* e *software*.

2.2 U-learning

Aprendizagem ubíqua, também conhecida como *U-learning* foi impulsionada pelo avanço da tecnologia e da comunicação de rede (JEONG e YI, 2014). Conforme Hooft (2006) a computação ubíqua quando relacionada à educação abrange os participantes ativos no processo de ensino-aprendizagem, pois possibilita que professores e alunos recebam as informações e criem conhecimentos novos de forma individual ou colaborativa. Para Rabello (2012), o paradigma da computação ubíqua promove a ideia da troca de informações a qualquer hora e em qualquer lugar, por meio da utilização, de forma transparente, inteligente e integrada às tecnologias computacionais. Hooft (2006) complementa que a computação ubíqua quando relacionada à educação

abrange os participantes ativos no processo de ensino-aprendizagem, pois possibilita que professores e alunos recebam a informação e criem conhecimentos novos de forma individual ou colaborativa.

Desta forma, diversas áreas de pesquisa que integram *U-learning*, entre elas pode-se citar a computação móvel e a ciência de contexto. O laboratório virtual é um ambiente que aborda características de *U-learning*, visto que realiza adaptações baseadas no contexto do usuário e no computacional. Para identificar as informações que interferem na sua utilização e realizar as adaptações necessárias nesse ambiente, foi-lhe integrado o Módulo Adaptativo. Essa integração foi realizada com intuito de evitar problemas abruptos relacionados com a baixa velocidade de conexão ou com os tipos de dispositivos, uma vez que, muitas vezes essas limitações limitam ou até impedem o estudante de carregar e visualizar os materiais corretamente. Esses entraves não são mais admitidos pelos usuários. Por fim, o estudo realizado nesta seção embasou o desenvolvimento deste trabalho, no qual procurou-se identificar e atender as características de ubiquidade apresentadas acima.

3 | METODOLOGIA

O objetivo principal desta pesquisa foi criar um ambiente de nuvem privada com características de um sistema *U-learning* e que, integrado ao Módulo Adaptativo, auxilie/facilite a aprendizagem de Redes de Computadores.

Para o desenvolvimento deste trabalho, foi empregada a pesquisa experimental, a qual tem por finalidade selecionar variáveis capazes de modificar o ambiente de estudo. Tendo em vista a natureza da pesquisa, esta é classificada como uma pesquisa aplicada, cujo propósito é buscar a solução de um objetivo específico. Primeiramente, foi definido o projeto de arquitetura, que contempla a infraestrutura de *hardware* necessária para o desenvolvimento desta pesquisa. A infraestrutura utilizada para realizar o experimento é composta por três máquinas, sendo: um servidor Dell Power Edge T300 com processador Intel Xeon Quadcore X3363 2.83 GHz com 4 núcleos físicos e 4 núcleos virtuais, memória RAM de 8 GB, 2 discos rígidos de 500GB e dois servidores com processador Intel Core2 Quad 2.88 GHz, contendo 4 núcleos físico-virtuais, memória RAM de 8GB e disco rígido de 500 GB.

A segunda etapa considera o problema, os objetivos e a contribuição pretendida com esta pesquisa. Nesta fase, foi definida e avaliada a plataforma de *Cloud Computing* OpenNebula. A linguagem de programação selecionada para a implementação do Módulo Adaptativo foi o PHP, por ser a linguagem nativa do AVA Moodle, e, para o desenvolvimento do medidor de velocidade de rede foi utilizada a linguagem JavaScript. A terceira fase consistiu na instalação das ferramentas que compõem o laboratório virtual U-Lab Cloud, sendo elas: WampServer4 versão 2.2 (inclui as ferramentas:

Apache 2.2.22; Mysql 5.5.24; PHP 5.3.13 e PhpMyadmin 3.4.10.1); Open Simulator versão 0.7.4; Moodle 6 versão 2.5; Sloodle 7 versão 2.1 for Moodle 2.5.

Na quarta etapa, foi realizada a inserção do Módulo Adaptativo. Esse Módulo tem por finalidade coletar e tratar as informações, como a velocidade de rede, tipo de dispositivo e o estilo cognitivo dos alunos. Na quinta etapa, foi definido o método de avaliação do U-Lab Cloud.

Para orientar a seleção dos métodos de avaliação utilizados nesta pesquisa, foi realizado um mapeamento sistemático sobre as avaliações de implementações de *Cloud Computing* para Ambientes Educacionais. Com base nos resultados obtidos, foi possível concluir que esse tema ainda é pouco explorado e que não existe um método bem definido para avaliação de sistemas educacionais disponibilizados em ambiente de nuvem.

Considerando o referido cenário, este trabalho adotou como método de avaliação as técnicas de simulação computacional, em razão de que a avaliação com estudantes reais necessita de um período grande de tempo e um número considerável de estudantes dispostos a colaborar. Conforme Iglesias et al. (2009), quando se trata de sistemas em avaliação é difícil de convencer uma quantidade suficiente de estudantes a participar desse tipo de experimento. Para isso, foi necessário realizar uma adaptação na ferramenta de simulação MADEA (Modelagem Automática e Dinâmica de Estilos de Aprendizagem), a qual foi desenvolvida por Dorça (2012) em sua tese. Essa ferramenta tem por objetivo detectar e corrigir de forma dinâmica e automática as inconsistências de estilos de aprendizagem do estudante.

4 | LABORATÓRIO VIRTUAL: U-LAB CLOUD

Para disponibilizar o U-Lab Cloud foi utilizada a *Cloud Computing* com o propósito de agregar a característica de elasticidade, facilitando o gerenciamento dos recursos computacionais de cada máquina virtual criada na nuvem privada, sendo possível gerenciar de forma dinâmica o funcionamento das máquinas virtuais de acordo com a necessidade de cada usuário. A outra contribuição pretendida com esta pesquisa é empregar as características da Computação Ubíqua ao laboratório virtual disponibilizado no ambiente de nuvem. Com isso, pretende-se auxiliar os estudantes a receber a informação adequada e no momento correto.

Nesse ambiente, a apresentação do material disponibilizado ao aluno receberá a adaptação de acordo com o perfil cognitivo, a velocidade de conexão e o tipo de dispositivo. Essas informações serão identificadas por meio do Módulo Adaptativo que está integrado ao AVA Moodle. O Módulo Adaptativo tornará o U-Lab Cloud dinâmico, uma vez que auxilia na identificação de variáveis de contexto e atende às características de mobilidade e disponibilidade.

A arquitetura proposta para o laboratório virtual foi projetada em três camadas:

Apresentação, Aplicação e Dados (Figura 1). A camada de Apresentação é a mais superficial deste ambiente e fornece a interface do ambiente. Para acessá-la, o usuário precisará ter instalado em seu dispositivo/computador apenas um *browser*, que permita a navegação na Internet para ter acesso ao laboratório virtual U-Lab Cloud.

A camada de Aplicação é a intermediária, sendo os seus principais componentes o TCN⁵ (Voss, 2014), o SEDECA (Mozzaquatro, 2010) e o U-SEA (Piovesan, 2011). Nesta camada, o laboratório virtual é composto pelas ferramentas OpenSim e Sloodle utilizadas no desenvolvimento do TCN⁵. Também encontra-se o Módulo Adaptativo, o qual foi uma integração do SEDECA e do U-SEA e tem por objetivo principal adaptar o ambiente de acordo com as características de contexto do estudante, e.g. a velocidade de conexão e o estilo cognitivo.

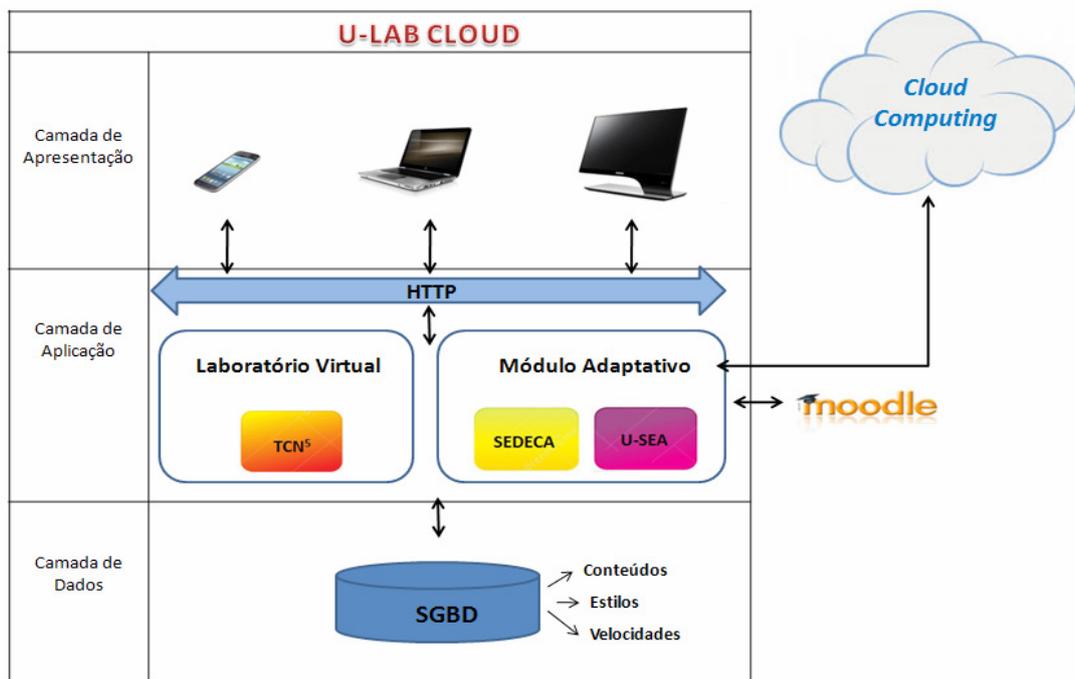


Figura 1. Arquitetura do U-Lab Cloud

O Módulo Adaptativo irá identificar essas informações por meio do SEDECA e do U-SEA, que foram integrados à plataforma Moodle. A identificação do estilo cognitivo será realizada pelo SEDECA, onde o usuário responderá um questionário para definir suas preferências em relação ao tipo de apresentação do conteúdo por exemplo, vídeo ou texto. Já o U-SEA irá verificar a velocidade de conexão e adaptará o ambiente considerando o tipo de material que é suportado com a velocidade de conexão identificada.

Por fim, o Módulo Adaptativo incorporado na Moodle se comunicará com o SGBD (Sistema Gerenciador de Banco de Dados) armazenado na camada Dados e adaptará a apresentação do conteúdo no laboratório virtual conforme com a preferência de cada estudante.

Ao contrário dos ambientes tradicionais, este dispensará a instalação, a manutenção ou a atualização de programas, pois estará sendo executado na nuvem

de forma transparente, ou seja, não apresentará nenhuma modificação aparente para o aluno, atendendo às características de um ambiente *U-learning*. A introdução dos elementos de *U-learning* como a mobilidade, possibilita o acesso ao ambiente educacional de forma mais eficiente para alunos e professores.

5 | IMPLEMENTAÇÃO

A implementação do U-Lab Cloud foi baseada no desenvolvimento de um ambiente de nuvem privada. Para isso, primeiramente foi necessário definir a plataforma de *Cloud Computing* mais adequada a esta pesquisa.

Assim sendo, dois aspectos principais foram considerados: o modelo de serviço e a infraestrutura. O laboratório virtual pertence ao modelo de serviço Infraestrutura como Serviço (IaaS), uma vez que fornece a infraestrutura para desenvolvimento de atividades de práticas do ensino de Redes de Computadores.

Assim, foi pesquisada, projetada e configurada a infraestrutura física para realização do experimento. O ambiente de testes utilizado é composto por três máquinas: na máquina principal foram atribuídas as funções de *cloud controle* (CC), *cluster controle* (CLC) e *storage controller* (SC); as outras duas máquinas receberam a função de *node controller*, as quais fornecem o processamento da nuvem.

Logo, foi implementada a plataforma de *Cloud Computing* OpenNebula. Para realizar a virtualização, foi selecionado o hipervisor VMware ESXi, por ser compatível com a plataforma de *Cloud Computing* selecionada. Com a virtualização é possível criar e administrar diversas máquinas virtuais no mesmo sistema operacional.

Assim, ao utilizar o recurso de virtualização garante-se níveis adequados de disponibilidade para o U-Lab Cloud, visto que caso a máquina virtual em que o ambiente estiver rodando falhe, o ambiente automaticamente será migrado para outra. Com isso, tem-se a intenção de manter o ambiente acessível, assegurando a sua disponibilidade. Dessa maneira, pode-se dizer que o laboratório virtual apresenta um dos elementos que compõe as características pretendidas para este ambiente de *Cloud Computing*.

Para a utilização do laboratório virtual no ambiente de nuvem privada, foi instanciada no VMware uma máquina virtual com sistema operacional Windows. Nela foi realizada a instalação das ferramentas instaladas no ambiente de *Cloud Computing* para criação do U-Lab Cloud. Na sequência, foi implantada a plataforma Moodle v. 2.5, onde foi criada a disciplina Redes de Computadores e foram inseridos alguns materiais e.g *slides* e vídeos.

O próximo passo foi a construção da interface do ambiente virtual 3D. Para isso, foi instalada a ferramenta OpenSim. Com o OpenSim e o Moodle em funcionamento, foi necessário integrá-los por meio da ferramenta Sloodle. Para isso, foi necessário copiar

as pastas dessa ferramenta para o diretório do Moodle. Os materiais disponibilizados no Moodle foram integrados com o laboratório virtual por meio das aplicações “*quiz chair*” e “*presenter*” disponibilizadas nesse aplicativo. Por fim, foi realizada a instalação do *viewer* Singularity, necessário para visualização da interface gráfica do U-Lab Cloud (Figura 2).



Figura 2. Ambiente U-Lab Cloud

Na figura 2, pode ser visualizado a interface gráfica do laboratório virtual no OpenSim. Dessa forma, o processo de implementação do U-Lab Cloud foi finalizado, onde pôde-se perceber diversas vantagens ao adotar a tecnologia de *Cloud Computing* em conjunto com as características *U-learning*. O baixo custo para o usuário é a principal delas, pois, para acessarem o laboratório virtual, não será necessário possuir tecnologia de ponta. Os usuários poderão acessar o ambiente de seus computadores pessoais (PCs), telefones celulares, *tablets*, mesmo que possuem uma configuração mínima, sendo necessário apenas que seus equipamentos estejam conectados a *Internet*.

Além disso, foi adaptada e integrada a plataforma Moodle a ferramenta de Modelagem de Estilos de Aprendizagem (MADEA) para realização dos testes. A principal finalidade de utilizar esta ferramenta de simulação computacional foi a possibilidade de obter uma visão geral sobre o funcionamento do ambiente, antes que o mesmo seja submetido a testes com estudantes reais.

6 | ANÁLISE DOS RESULTADOS

O estudo de caso simulado foi baseado na disciplina de Redes de Computadores disponibilizada na plataforma Moodle. A disciplina utilizada neste estudo de caso é real, contudo, o acompanhamento e os estudantes foram simulados com o auxílio

da ferramenta MADEA. Para empregá-la neste trabalho, foi necessário realizar a adaptação e a integração dela com a plataforma Moodle.

Para demonstrar a integração das ferramentas mencionadas, foi definido um estudante fictício com a seguinte preferência: ativo (forte), sensitivo (forte), visual (moderada) e sequencial (fraca). Com os resultados obtidos nesta simulação, foi possível observar a atualização dos estilos de aprendizagem do estudante fictício no decorrer da disciplina de Redes de Computadores, com intervalos de 10 iterações. Esta simulação computacional considerou modelos probabilísticos, onde as probabilidades são determinadas pelo valor de cada preferência dos estilos de aprendizagem.

A Figura 3 mostra a dimensão de Processamento, que representa como o estudante com forte preferência pelo estilo Ativo procede em relação ao processamento de informação. Esse estudante com preferência por uma experimentação ativa, possui preferência por trabalhos em pequenos grupos e que envolvam discussão durante o andamento da disciplina.

Na Figura 3, foi possível observar como evoluiu o estilo de aprendizagem probabilístico do estudante com preferência pela experimentação ativa. Assim, percebe-se que a preferência pelo estilo Ativo (forte) foi mantida durante todo o processo de aprendizagem. Entretanto, observa-se que em dois curtos espaços de tempo por volta de (05 e 30 iterações) o estilo de aprendizagem apresentou-se inconsistente, mas foi corrigido a medida que a quantidade de iterações cresceu.

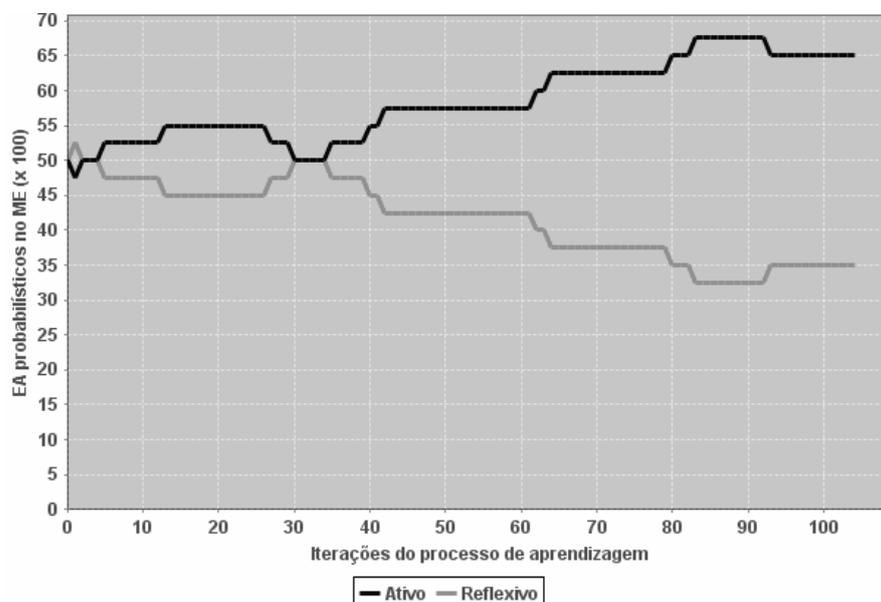


Figura 3 - Dimensão de Processamento.

A atualização da dimensão Percepção pode ser visualizada na Figura 4. Essa dimensão diz respeito a qual tipo de informação é melhor percebida pelo estudante. O estudante com forte preferência pelo estilo Sensitivo está propenso a encontrar no ambiente educacional os exemplos do mundo real que disponibilizam datas, dados experimentais e informações que podem ser percebidas pelos seus sentidos (ver, ouvir).

Na Figura 4, ao analisar o progresso do estilo de aprendizagem Sensitivo (forte), pôde-se constatar que durante a trajetória tal estilo manteve-se consistente. Após um terço do caminho percorrido (por volta de 30 iterações), percebe-se indícios de que o estilo de aprendizagem tornou-se mais estável.

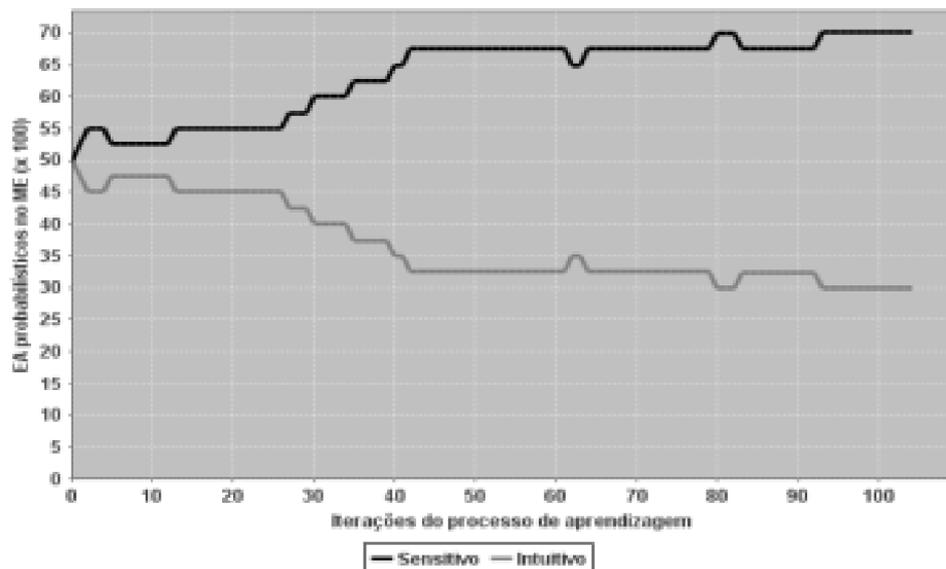


Figura 4 - Dimensão de Percepção.

Na Figura 5, é demonstrada atualização da dimensão Entrada. Essa dimensão define qual o tipo de informação é efetivamente percebida pelo estudante que apresenta preferência fraca pelo estilo Visual. Esse estilo é mais propenso a percepção de figuras, diagramas, linhas de tempo, gráficos e filmes.

A Figura 5 mostra os valores assumidos pelo estilo de aprendizagem Visual (fraco) durante o processo de aprendizagem simulado. Nota-se indícios de que o estilo de aprendizagem manteve-se instável até aproximadamente metade do processo por volta de 41 iterações). Após, percebe-se que o estilo tornou-se consistente.

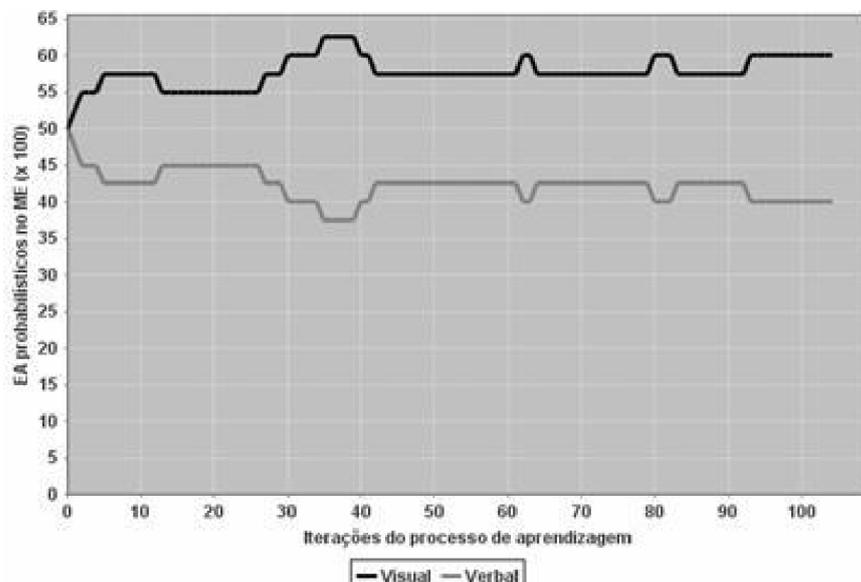


Figura 5 - Dimensão de Entrada.

Na Figura 6, é possível verificar a evolução da dimensão Organização. Essa dimensão determina como o estudante prefere progredir dentro da disciplina, sendo que neste caso, o estudante fictício apresenta fraca preferência pelo estilo de aprendizagem Sequencial. Dessa forma, ele está propenso a seguir uma progressão lógica e linear do conteúdo, ou seja, tem preferência por uma sequência de passos para solucionar problemas. A Figura 6 apresenta graficamente indícios de que durante o processo de aprendizagem, o estilo de aprendizagem mostrou-se bem definido, mesmo apresentando força de preferência fraca no modelo inicial de estudante.

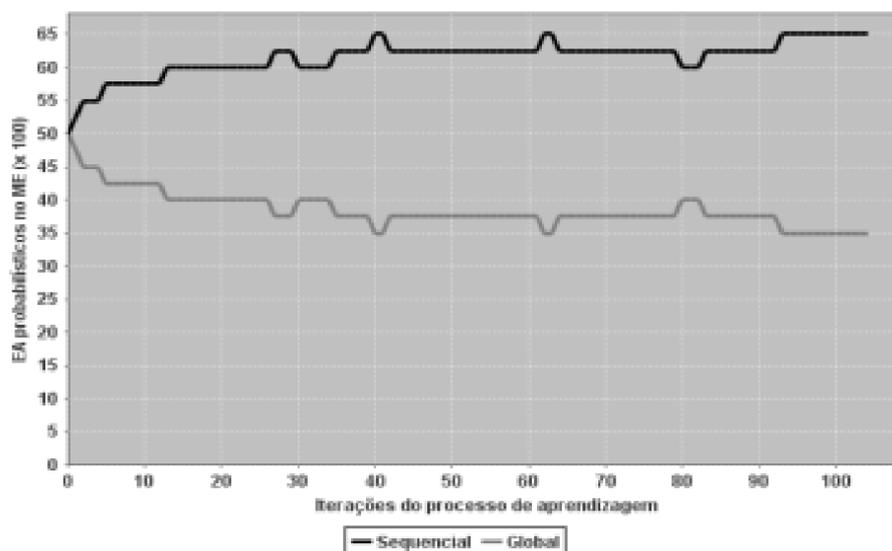


Figura 6 - Dimensão de Saída

O estudo de caso realizado demonstrou-se condizente com o objetivo desta abordagem, que buscou verificar o comportamento do estudante e do ambiente. A utilização de técnicas de simulação foi significativa para a avaliação deste trabalho, uma vez que propiciou executar testes em pouco tempo, o que seria dificultoso e mais complexo de realizar com estudantes reais. Dessa forma, acredita-se com os resultados desta avaliação, o U-Lab Cloud encontra-se apto para ser utilizado por estudantes reais em um processo de ensino-aprendizagem na disciplina de Redes de Computadores.

7 | CONSIDERAÇÕES FINAIS

Com relação ao estudo de caso desenvolvido, foi possível concluir que a utilização de técnicas de simulação para avaliar o funcionamento de um sistema educacional antes de submetê-lo a testes com estudantes reais foi fundamental, devido às limitações encontradas nesta etapa. Pode-se destacar como a principal limitação evidenciada em processos de avaliação de ambientes educacionais, a fragilidade das respostas obtidas por meio de questionários. A utilização desse recurso por estudantes

exige muito tempo e concentração, pois geralmente esse instrumento é extenso e os estudantes não estão a par da importância das suas respostas. Por isso, os estudantes acabam respondendo sem ter o cuidado de analisar o impacto do seu feedback, o que pode tornar os resultados pouco relevantes, até mesmo incompatíveis com os dados reais dos estudantes.

Além disso, outra limitação percebida foi a dificuldade de disponibilização de horários de aula em cursos reais, visto que os professores seguem suas atividades conforme seus planejamentos, não sendo possível realizar adaptações, uma vez que, trabalham com um calendário acadêmico pouco flexível. E, ainda, existe a preocupação com a exposição de informações de alunos e professores nas instituições de ensino.

Neste cenário, a avaliação com técnicas de simulação se mostrou eficiente, pois forneceu uma visão geral de forma contínua ao longo do processo de ensino-aprendizagem de Redes de Computadores. Com isso, pôde-se observar de forma mais precisa, como os estilos de aprendizagem sofrem alterações no decorrer do processo de ensino-aprendizagem. Desse modo, é possível identificar qual conteúdo é mais apropriado as necessidades individuais do aluno no momento certo, o que é um elemento fundamental para ambientes U-learning.

Esse trabalho apresentou resultados positivos com um grau mais preciso se comparado a outros que utilizam instrumentos de coleta de dados determinar o perfil do estudante. Tal abordagem foi realizada com baixo custo computacional proporcionado pela tecnologia de Cloud Computing, dado que foi reutilizado o hardware existente no laboratório de pesquisa GRECA/UFSM. Portanto, observou-se que as informações obtidas nesse trabalho podem apoiar a construção de ambientes adaptativos como o laboratório virtual U-Lab Cloud. Este ambiente se adapta ao contexto computacional do estudante, visto que verifica a velocidade de conexão, o tipo de dispositivo e o estilo cognitivo do aluno.

REFERÊNCIAS

Bai, Y.; Shen, S.; Chen, L.; Zhuon, Y. **Cloud learning: A new style**. In: IEEE International Conference on Multimedia Technology (ICMT), 2011. Anais... IEEE, p. 3460-3463, 2011.

Caminero, A. et al. **Obtaining university practical competences in engineering by means of virtualization and cloud computing technologies**. In: IEEE International Conference on Teaching, Assessment and Learning for Engineering (TALE), 2013. Anais. IEEE, 2013. p. 301-306.

Dey, A. K. **Understanding and Using Context**. *Personal and Ubiquitous Computing*, [S.l.], v.5, n.1, p.4-7, 2001.

Doelitzscher F. et al. **Private cloud for collaboration and e-Learning services: From IaaS to SaaS**. *Journal Computing*, [S.l.], v.91, p. 23-42, 2011.

Dorça, F. A. **Uma abordagem estocástica baseada em Aprendizagem por Reforço para modelagem automática e dinâmica de Estilos de Aprendizagem de Estudantes em Sistemas Adaptativos e Inteligentes para Educação a Distância**. Tese. (Doutorado em Ciências) - Faculdade de Engenharia Elétrica da Universidade Federal de Uberlândia. 2012.

Fillipetti, M. A. **Uma arquitetura para a construção de laboratórios híbridos de redes de computadores remotamente acessíveis**. Dissertação (Mestrado em Engenharia da Computação) – Instituto Pesquisas Tecnológicas do Estado de São Paulo (IPT). 2008.

Herpich, F.; Nunes, F. B.; Voss, G. B.; Medina, R. D. **Three-Dimensional Virtual Environment and NPC: A Perspective about Intelligent Agents Ubiquitous**. In F. M. Neto, R. de Souza, & A. S. Gomes (Eds.), *Handbook of Research on 3-D Virtual Environments and Hypermedia for Ubiquitous Learning* (pp. 510–536). Hershey PA, USA: IGI Global. 2016.

Hooft, M. **What is ubiquitous computing?** In: Hooft and Swan, *Ubiquitous Computing in Education Invisible technology, visible impact*, London: Lawrence Erlbaum Ass., p.3-17, 2006.

Iglesias, A. **Learning teaching strategies in an adaptive and intelligent educational system through reinforcement learning**. *International Journal Applied Intelligence*, [S.l.], v.31, n.1, p. 89–106, 2009.

Jadela Y.; Modi K. **Cloud computing - concepts, architecture and challenges**. In: IEEE International Conference on Computing, Electronics and Electrical Technologies (ICCEET). Kumaracoil, p. 877 - 880, 2012.

Jeong, H-Y.; Yi, G. **A Service Based Adaptive U-Learning System Using UX**. In: *The Scientific World Journal*, [S.l.], 2014.

Mell, P.; GRANCE, T. **The Nist Definition of Cloud Computing**. In: *National Institute of Standards and Technology*, [S.1.], p. 7, 2011.

Mozzaquatro, P. M. **Adaptação do Mobile Learning Engine Moodle (MLE MOODLE) aos diferentes estilos cognitivos utilizando Hipermedia Adaptativa**. Dissertação (Mestrado em Computação) Universidade Federal de Santa Maria. 2010.

Orgaz, G. B. et al. **Clustering Avatars Behaviours From Virtual Worlds Interactions**. In: *International Workshop On Web Intelligence & Communities - WI&C'12*, 4., New York, USA. Proceedings. ACM Press, p.1, 2012.

Piovesan, S. **U-SEA: um ambiente de aprendizagem ubíquo utilizando cloud computing**. Dissertação (Mestrado em Computação) - Universidade Federal de Santa Maria. 2011.

Rabello S. et al. **Um Modelo para Colaboração em Ambientes Descentralizados de Educação Ubíqua**. Rio de Janeiro, RJ: Anais. 23º Simpósio Brasileiro de Informática na Educação (SBIE), v.23, n.1, 2012.

Voss, G. TCN⁵ - **Teaching Computer Networks In a Free Immersive Virtual Environment**. Dissertação (Mestrado em Computação) - Universidade Federal de Santa Maria. 2014.

Yahya, S.; Ahmad, E. A.; Jalil, K. A.; Mara, U. T. **The definition and characteristics of ubiquitous learning : A discussion**. *International Journal of Education and Development using Information and Communication Technology*, v. 6, p. 1–12, 2010.

Wang X.; Hembroff C. G.; Yedica R. **Using VMware VCenter lab manager in undergraduate education for system administration and network security**. In: *Proc. of The 11th acm Conference on Information Technology Education (SIGITE)*, (Midland, USA), 2010.

ANÁLISE DA APRENDIZAGEM DE PROGRAMAÇÃO POR MAPEAMENTO DE PERFIS EM MÉTRICAS DE SOFTWARE

Márcia Gonçalves de Oliveira

Instituto Federal do Espírito Santo (Ifes)
Vitória - ES

Ádler Oliveira Silva Neves

Instituto Federal do Espírito Santo (Ifes)
Vitória – ES

Helen França Medeiros

Instituto Federal do Espírito Santo (Ifes)
Vitória – ES

Mônica Ferreira Silva Lopes

Instituto Federal do Espírito Santo (Ifes)
Vitória – ES

Leonardo Leal Reblin

Universidade Federal do Espírito Santo (Ufes)
Vitória - ES

Elias Silva de Oliveira

Universidade Federal do Espírito Santo (Ufes)
Vitória – ES

RESUMO Este trabalho apresenta o *PCódigo II*, um sistema de mapeamento automático de perfis de estudantes em métricas de software para análise da aprendizagem de programação. Além do mapeamento de perfis em 348 métricas de software, o *PCódigo II* possui as funcionalidades de execução em massa, de agrupamento de perfis similares, de visualização da informação e de análise de plágios. As primeiras aplicações do *PCódigo II* em exercícios reais de cursos de programação

a distância mostram que professores, atentando para o que as métricas informam, podem reconhecer as dificuldades de aprendizagem, boas práticas de programação e classes de perfis de aprendizagem de toda uma turma de forma rápida, detalhada e holística.

PALAVRAS-CHAVE análise de aprendizagem, métricas de software, programação.

ABSTRACT This work presents *PCódigo II*, a system of an automatic mapping of student profiles in software metrics to analyze programming learning. In addition to profile mapping in 348 software metrics, *PCódigo II* has mass execution, similar profile grouping, information visualization, and plagiarism analysis capabilities. The first applications of *PCódigo II* in real programming exercises demonstrate the effectiveness of this system for the diagnostic evaluation of programming learning. The first applications of *PCódigo II* in real programming exercises show that teachers, taking into account what the metrics say, can recognize the learning difficulties, good programming practices and classes of learning profiles of a whole class in a fast, detailed and holistic way.

KEYWORDS learning analysis, software measures, programming.

1 | INTRODUÇÃO

A avaliação da aprendizagem de programação com as finalidades de diagnosticar, regular e qualificar um processo de aprendizagem tem sido um verdadeiro desafio, uma vez que a análise de aprendizagem de programação é, de modo geral, baseada em indicadores subjetivos. Dessa forma, a carência de indicadores padronizados inviabiliza uma avaliação para reconhecimento de dificuldades de aprendizagem, habilidades e até competências em programação. Por conseguinte, a automatização desse processo também é dificultada.

Embora já existam soluções de avaliação automática de programação (PIETERSE, 2013), uma importante questão a ser discutida é a seguinte: em vez de aplicar testes padronizados, acatar a avaliação subjetiva de professores ou buscar na verbalização de estudantes possíveis indicadores de aprendizagem, não seria mais viável uma avaliação a partir da análise dos códigos de programação sob diferentes métricas para inferir dificuldades, habilidades e competências em programação?

Uma alternativa para essa avaliação é o uso de métricas de software para análise de códigos-fontes visando quantificar esforço e qualidade de programação (CURTIS *et al.*, 1979, BERRY; MEEKINGS, 1985). Essas métricas podem ser utilizadas, conforme Pettit *et al.* (2015), como instrumentos de análise dos processos de programação a partir de códigos-fontes para auxiliar professores na avaliação de seus alunos.

Com o objetivo de analisar a aprendizagem de programação a partir de códigos-fontes para reconhecimento de dificuldades de aprendizagem, habilidades e até competências em programação, foi desenvolvido o *PCódigo II*, uma evolução do sistema *PCódigo* desenvolvido por Oliveira (2015) na sua forma de mapear perfis que consiste em representar soluções de programação desenvolvidas por estudantes em 348 métricas de software.

A principal contribuição deste trabalho para a Informática na Educação é propor um instrumento de apoio à avaliação que possibilite professores realizarem uma análise multidimensional da aprendizagem de seus alunos na prática da programação. Dessa forma, através de um amplo leque de métricas, professores poderão identificar classes de perfis de alunos, analisar indicadores de possíveis causas de dificuldades de aprendizagem e comparar soluções de programação em detalhes.

Para apresentar os fundamentos e funcionalidades do *PCódigo II*, este trabalho está organizado conforme a ordem a seguir. A Seção 2 apresenta os trabalhos relacionados. A Seção 3 descreve a arquitetura do *PCódigo II* e as principais métricas de software utilizadas na representação de perfis. Na Seção 4, destacam-se a aplicação do *PCódigo II* em cursos a distância de programação e os principais resultados. A Seção 5 conclui este trabalho destacando os principais achados, os trabalhos futuros e as considerações finais.

2 | TRABALHOS RELACIONADOS

Os principais trabalhos relacionados à proposta deste trabalho são os trabalhos de Oliveira *et al.* (2013), de Munson (2017) e de Oliveira (2015). O trabalho de Oliveira *et al.* (2013) mapeia informações de código C em componentes de habilidades. Além dessas componentes de habilidades, as métricas de avaliação de classificação automática *multi-label* foram utilizadas por Oliveira *et al.* (2013) como instrumentos de avaliação diagnóstica fornecendo importantes indicadores de dificuldades de aprendizagem a serem acompanhados por professores.

Um estudo mais recente visa encontrar métricas para determinar, no início de um curso, quais alunos podem estar em risco (MUNSON, 2017). Nessa proposta, os *logs* de atividades de programação gerados por um ambiente de programação foram utilizados para gerar um conjunto de dados de variáveis, como o tempo e a quantidade de erros.

O *PCódigo* de Oliveira (2015), por sua vez, é um sistema de apoio à prática assistida de programação que recebe soluções de programação submetidas por estudantes via interface web, executa-as e emite relatórios de avaliação para professores. As principais funcionalidades do *PCódigo* para apoiar o trabalho docente e favorecer a aprendizagem de programação são as seguintes (OLIVEIRA, 2015): executar programas em massa, representar perfis em componentes de habilidades e analisar códigos-fontes.

O *PCódigo II* herda as funcionalidades do *PCódigo* original de Oliveira (2015), evoluindo-o na representação de perfis e, conforme Oliveira *et al.* (2013), dá novos significados a diferentes métricas para avaliação diagnóstica da aprendizagem de programação visando identificar, assim como Munson (2017), alunos com dificuldades de aprendizagem.

3 | O PCODIGO II

O *PCódigo II* foi desenvolvido com a finalidade de ser um sistema de análise multidimensional da aprendizagem de programação a partir da análise de códigos-fontes escritos por estudantes. Para isso, cada programa escrito caracterizando um perfil foi representado por um vetor de 348 dimensões, onde cada dimensão representa uma métrica de software.

A ideia de utilizar métricas de software na representação de perfis de estudantes de programação surgiu da necessidade de quantificar o trabalho de programação realizado através de variáveis de avaliação que indicassem, por exemplo: esforço de programação, complexidade de código, estilo de programação, número de variáveis, número de linhas, número de comentários, eficiência, complexidade e número de

funções.

O *PCódigo II* estende o *PCódigo* original de Oliveira (2015) na representação de perfis de estudantes pelas métricas de análise de códigos-fontes em Linguagem C (BERRY; MEEKINGS, 1985), pelas métricas de *Halstead* e *McCabe* (CURTIS *et al.*, 1979) e pelos indicadores de execução *compila* e *executa* de Oliveira (2015). Ao todo, o *PCódigo II* implementa 348 métricas para a caracterização de perfis de estudantes. A Tabela 4 apresenta as principais métricas do *PCódigo II*.

A Figura 1 apresenta a arquitetura do *PCódigo II* com suas funcionalidades, processos, entradas e saídas.

De acordo com a Figura 1, o processo de mapeamento de perfis em métricas de software começa com a Submissão de uma Solução em C junto com um arquivo makefile e um arquivo entrada. A Solução em C contém um ou mais arquivos de um projeto contendo programas em Linguagem C, arquivos de bibliotecas e outros arquivos necessários para rodar um programa. O arquivo makefile contém as instruções de execução da Solução em C e o arquivo entrada contém as entradas utilizadas para testar a Solução em C.

Cada Submissão é armazenada em uma base de Exercícios de Programação via interface web do MOODLE na Visão Aluno, conforme a Figura 1. Todas as submissões para cada tarefa são executadas pelo Núcleo Executor do *PCódigo II* e os arquivos executável e arquivos saída com os resultados de execução de uma Solução em C são gerados em cada diretório de Submissão.

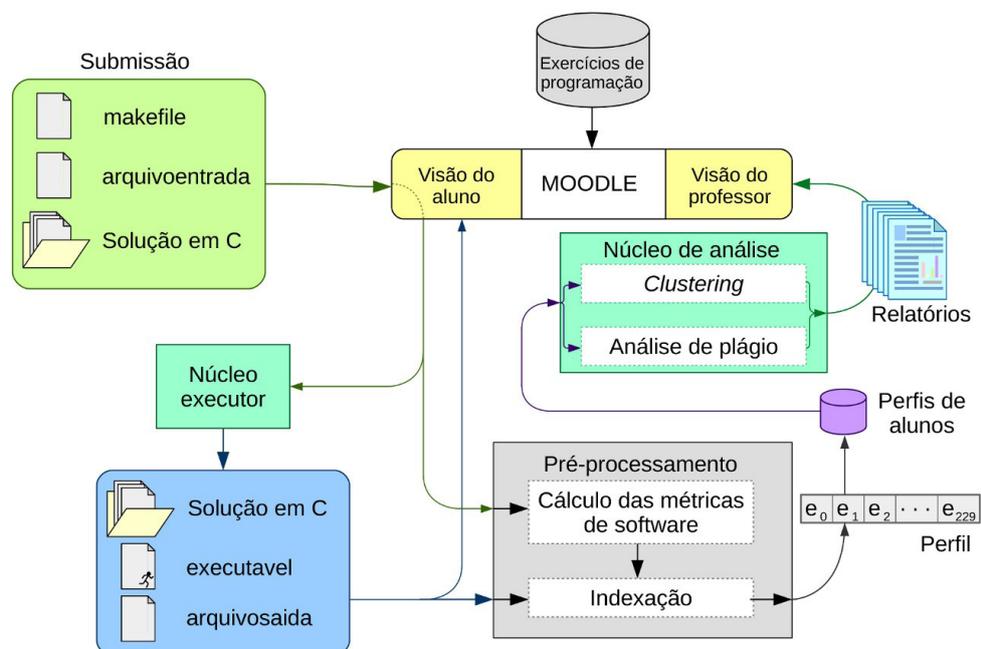


Figura 1. Arquitetura do *PCódigo II* (NEVES *et. al.*, 2017)

O *Pré-processamento* da Figura 1 consiste em duas etapas: *Cálculo das Métricas* e *Indexação*. O *Cálculo das Métricas* consiste em analisar os códigos de programação

em Linguagem C e a partir deles calcular os valores das métricas de Curtis *et al.* (1979) e de Berry e Meekings (1985) e dos indicadores de execução de Oliveira (2015). A *Indexação* consiste em criar uma representação vetorial chamada *Perfil* em que cada dimensão contém o valor e_i ($i = 1 \dots 348$) de uma métrica de software calculada. A Figura 2 apresenta algumas métricas utilizadas na representação de perfis de estudantes de programação.

Em seguida, conforme a Figura 1, o Perfil gerado na *Indexação* é armazenado junto com outros perfis da mesma tarefa na Matriz M (OLIVEIRA, 2015). Essa matriz é normalizada a valores entre 0 e 1 e submetida a algoritmos de Clustering para agrupamento de perfis similares e ao algoritmo de Análise de Plágio para verificar similaridades entre perfis que caracterizem plágios.

Finalizando os processos, na Figura 1, os Relatórios de Clustering, de Visualização da Informação e de Análise de Plágio, que são descritos nas subseções a seguir, são enviados para um professor via interface web *Visão do Professor* do Moodle.

Através dos *Relatórios do PCódigo II*, os professores podem analisar a aprendizagem de seus alunos comparando perfis e reconhecendo, por meio das métricas, dificuldades de aprendizagem, boas práticas de programação e plágios.

3.1 Clustering e visualização da informação

Para agrupar os perfis por similaridade, foi utilizado o algoritmo de *clustering Bisecting K-means* do software *Cluto* (KARYPIS, 2002). As entradas foram o número de *clusters*, a matriz M' formada pelos vetores de perfis, os rótulos das linhas de M' com os identificadores das soluções de cada aluno e os rótulos das colunas com os identificadores das métricas de software.

Métrica	Origem	Significado
compila	PCodigo	programa escrito corretamente
executa	PCodigo	O código executa?
article_berry_reserved_words	ref:RBerry85	o número de diferentes palavras reservadas e
article_curtis_halstead_difficulty	ref:BCurtis79	Dificuldade de Halstead
article_curtis_halstead_effort	ref:BCurtis79	Esforço de Halstead
article_curtis_halstead_volume	ref:BCurtis79	Volume de Halstead
function_count	git:lzd	Quantidade de funções utilizadas
halstead_difficulty	git:ccd	Dificuldade de Halstead
halstead_effort	git:ccd	Esforço de Halstead
token_count	git:ccd@	Número de símbolos utilizados em todo o programa
lines_of_code_total	git:ccd@	Contagem de linhas do código fonte.
reserved_word_for	git:ccd@	Símbolo reservado pela gramática da linguagem
reserved_word_if	git:ccd@	Símbolo reservado pela gramática da linguagem
cyclomatic_complexity_avg_per_func	git:lzd	Média da complexidade ciclômática de todas as funções.
Abreviações Prefixos ref: referencia a bibliografia em BibTeX, abaixo git: referencia a saída de um código vindo do GitHub Corpo ccd: https://github.com/dborowiec/commentedCodeDetector/tree/master lzd: https://github.com/terryyin/lizard/tree/master Sufixo @: indica que o código recebeu edições para desempenhar tal funcionalidade		

Figura 2. Exemplos de métricas de software (NEVES et. al, 2017)

As saídas dos algoritmos de *clustering* foram um arquivo contendo a identificação dos *clusters* a que cada solução da matriz M' pertence, um gráfico de *clustering* com os vetores de soluções distribuídos entre os *clusters* e um relatório com as informações de cada *cluster* e dos processos de *clustering*.

Para a visualização de informação, foram utilizados algoritmos escritos em Linguagem R para geração de mapas de calor (KOLDE, 2015). Os algoritmos de *clustering* do *Cluto 2.1.2* (KARYPIS, 2002) geraram uma visualização de perfis reunidos em *clusters*. A Figura 4 é um exemplo de gráfico gerado pelo *Cluto*.

Os *clusters* foram obtidos usando a medida de similaridade coeficiente de correlação. Dessa forma, os valores das métricas no gráfico gerado correspondem aos valores do vetor original subtraídos do vetor-média.

3.2 Análise de plágios

No módulo de *Análise de Plágio* da Figura 1, os vetores da Matriz M' são comparados dois a dois e é gerada uma Matriz M_A , formada pelos índices de similaridade entre cada par de vetores (OLIVEIRA, 2015) calculados pela medida *cosseño* (BAEZA-YATES et al., 1999). Os índices de similaridade variam de 0 (dissimilaridade) a 1 (similaridade total).

Para a análise do professor, o módulo *Análise de Plágio* retorna os pares de

vetores com índices de similaridade acima de 0.9, isto é, com semelhanças acima de 90% (OLIVEIRA, 2015), conforme exemplo da Figura 5.

A vantagem do *PCódigo II* em relação ao *PCódigo* original na *Análise de Plágio* é não precisar realizar processos de normalização de códigos para retirada de *tokens* que geram ambiguidades, de *strings* e de comentários, uma vez que, comparando soluções a partir de 348 variáveis de avaliação, as similaridades acima de 90% evidenciam fortemente práticas de plágios.

4 | EXPERIMENTOS E RESULTADOS

Para coletar códigos de programação e para testar as funções de representação de perfis em métricas de software, de *clustering*, de visualização da informação e de análise de plágios do *PCódigo II*, foi ofertado um curso a distância de programação C para 80 estudantes do ensino médio, de graduação e de pós-graduação.

Durante o curso, foi aplicado um exercício de programação para avaliação diagnóstica, o mesmo utilizado por Oliveira (2015). Esse exercício é adequado para a avaliação diagnóstica porque utiliza expressões lógicas, estruturas de controle condicional e estruturas de controle de repetição. Para o professor que aplicou a atividade, o critério de avaliação foi o uso das expressões lógicas. Dessa forma, seria uma evidência de dificuldades o uso excessivo de comparações e um número alto de linhas de código. Uma boa solução utilizaria no máximo três comparações e poucas linhas de código.

O experimento foi realizado com trinta alunos que submeteram uma solução em código C para o exercício proposto por meio do ambiente virtual *Moodle*. Essas soluções submetidas foram executadas e mapeadas em vetores de 84 dimensões correspondentes às métricas de software não nulas para todos os alunos. Em seguida, foi gerada uma visualização desses vetores, que é apresentada na Figura 3.

Ao contrário das soluções *A*, *E* e *C*, a Solução *D* do gráfico da Figura 3 evidencia dificuldades de aprendizagem pelas taxas muito baixas ou nulas nos valores de várias métricas. Nesse caso, o predomínio da cor azul, em especial nas métricas de *Dificuldades*, indica dificuldades pela ausência de instruções de programação. Ao verificar essa solução, observou-se que o aluno só escreveu instruções básicas de entrada e saída, não utilizando expressões lógicas nem estruturas de controle condicional e de repetição. O aluno não conseguiu, portanto, operar logicamente, o que é grave, uma vez que as expressões lógicas são conteúdos fundamentais na aprendizagem de programação.

A Figura 4 apresenta a visualização gerada pelos algoritmos de *clustering*. As linhas são rotuladas pelos identificadores dos alunos, *underline* e a nota do aluno com valores de 0 a 1 (0=0% e 1=100%). Quanto mais vermelho estiver o valor de uma métrica, maior é esse valor em relação ao valor médio da métrica e, quanto mais verde, menor é o valor dessa métrica em relação ao seu valor médio. A cor preta indica valor zero, isto é, a métrica assume o valor médio.

De acordo com o gráfico da Figura 4, as soluções plagiadas *P* aparecem no mesmo *cluster*. No entanto, a alta similaridade entre os dois códigos passou despercebida na avaliação do professor, uma vez que uma solução obteve nota 0.6 (60%) e a outra 1.0 (100%). Além disso, conforme o predomínio da cor vermelha nas métricas de complexidade e de dificuldade do gráfico, as duas soluções indicam dificuldades de aprendizagem por excesso de código. Através desse tipo de visualização, podemos ver a importância do uso das métricas para auxiliar o trabalho de avaliação de professores de programação.

Uma outra importante observação no gráfico da Figura 4 aparece nas soluções da parte mais baixa do gráfico. De forma isolada, aparecem as soluções *A*, *C*, *D* e *E*, isto é, as soluções que evidenciaram mais dificuldades de aprendizagem. As soluções *A* e *E*, segundo critérios do professor, obtiveram nota máxima e as soluções *C* e *D* obtiveram notas 0.0 e 0.4, respectivamente. As soluções *A* e *E*, embora indicassem muito esforço, o professor possivelmente considerou que a solução “deu certo” embora não “estivesse certa”. A Solução *C* possivelmente recebeu nota zero por não ter compilado e a Solução *D* perdeu 0.6 porque estava incompleta sem as expressões lógicas e sem as estruturas condicionais e de repetição.

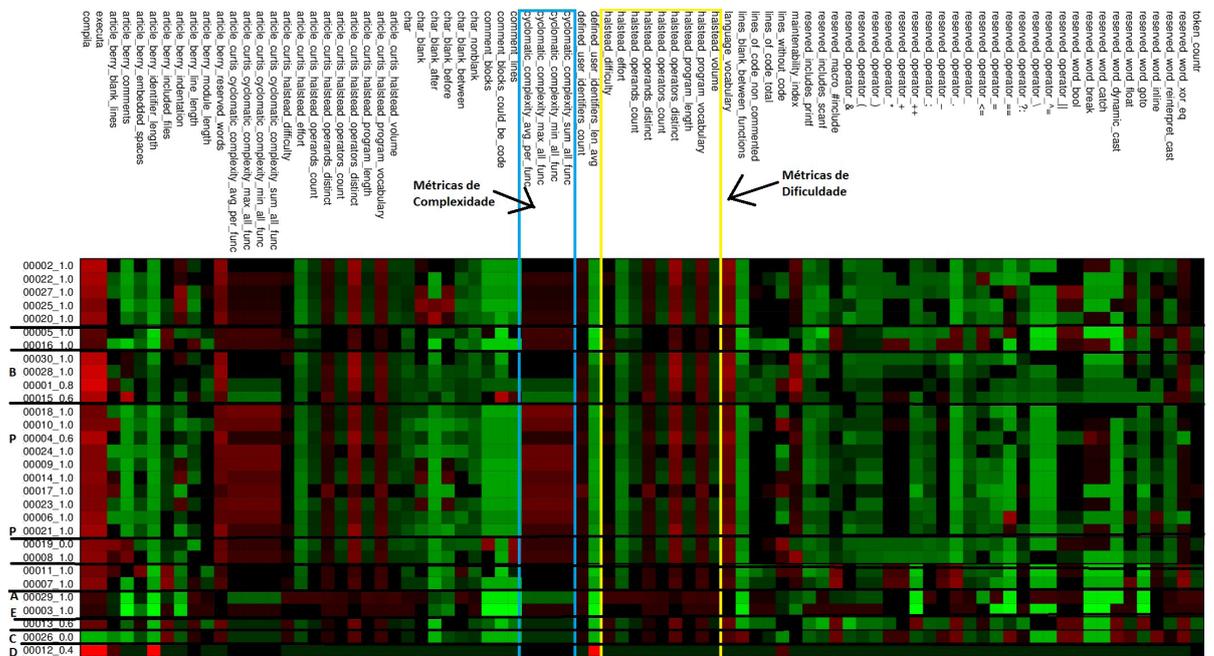


Figura 4. Gráfico de clustering

A *Solução B* e as demais soluções presentes no mesmo *cluster* na Figura 4 aparecem com predomínio de baixos valores de complexidade e a *Solução B* e a solução abaixo dela aparecem com predomínio da cor preta nas métricas de complexidade, confirmando, por estarem com os valores médios dessas métricas, que são as melhores soluções da turma, embora não estejam 100% corretas conforme análise de um professor de programação.

A Figura 5 apresenta o relatório de similaridade das soluções submetidas reconhecidas como suspeitas de plágio, isto é, com similaridade acima de 90%. Nas linhas 19 e 20 do relatório, aparecem as soluções *P*, chamadas, respectivamente, de *Solução 4* e *Solução 21*. Essas soluções têm similaridade de 97.78%, indicando fortes indícios de plágio. Analisando os vetores das linhas 19 e 20, observamos que os valores normalizados das métricas eram muito próximos. A Tabela 1 apresenta os códigos dessas soluções e confirma a alta similaridade entre elas.

De acordo com a Tabela 1, os códigos suspeitos de plágio assemelham-se pelas instruções e pelo erro comum de utilizar a estrutura de seleção da Linguagem C *switch*. Conforme Oliveira (2015), quando os alunos têm dificuldades em programar e resolvem "colar", eles costumam mudar apenas os nomes dos identificadores como, por exemplo, os nomes de variáveis. No entanto, eles não alteram os códigos de programação por não entendê-los. Observando o início das duas soluções da Tabela 1, conclui-se que o plagiador fez a alteração apenas nos nomes das variáveis.

Solução 4	Solução 21
<pre> #include <stdio.h> int main () { int t, gn, vf, gp, vc, e, pt1, p t2, pt3, camp, vice; for(t=1;t<=3;t++) { printf (@texto, t); printf (@texto); scanf ("%d", &gp); ... switch(t){ ca se 1: pt1 = (5*gp - gn + 3*vf + 2*vc + e); pri ntf (@texto, t, pt1); break; ... case 3: pt3 = (5*gp - gn + 3*vf + 2*vc + e); pri ntf (@texto, t, pt3); break; default:br eak; } } if(pt1>pt2) camp = 1; else camp = 2; if(pt3>camp) camp = 3; ... if (camp == 3) if (pt2>pt1) vice = 2; else vice = 1; printf (@texto, camp, vice) } </pre>	<pre> #include<stdio .h>int main () { int time, GP, GN, VF, VC, E, P1, P2, P3, primeiro, seg undo; for(time=1;time<=3;time++) { printf (@texto, time); printf (@text o);scanf ("%d" ,&GP); ... switch(time){ case 1: P1=(5*GP - GN + 3*VF + 2*VC + E); printf (@texto, time, P1); break; ... case 3: P3=(5*GP - GN + 3*VF + 2*VC + E); printf (@texto, time, P3); break; default:break; }} if(P1>P2) primeiro =1;else primeiro = 2; if(P3>prime iro)primeir o = 3; ... if (primeiro ==P3)if (P2>P1) segundo =2;else segundo = 1; printf (@texto, primeiro, segun do);return 0; } </pre>

Tabela 1. Programas suspeitos de plágios

Os resultados apresentados demonstram que a análise de aprendizagem por métricas de software é uma possibilidade muito favorável ao trabalho de avaliação de professores de programação, pois auxiliam-nos a compreenderem as dificuldades de aprendizagem de seus alunos principalmente quando estas se evidenciam em uma turma inteira. Além disso, as métricas podem ajudar professores a descobrirem o que caracteriza as boas soluções.

B	C	D	E	F	G	H
*****		PCodigo	-	RELATÓRIO	DE	ANÁLISE
Variáveis	compila	executa	nota	article_berry_blank_lines	article_berry_comments	article_berry_constant_definitions
al_00014	1.0000	1.0000	1.0000	0.5517	0.0000	0.0000
al_00023	1.0000	1.0000	1.0000	0.1724	0.0000	0.0000
Similaridade:	0.985808	(98.5808%)				
Variáveis	compila	executa	nota	article_berry_blank_lines	article_berry_comments	article_berry_constant_definitions
al_00009	1.0000	1.0000	1.0000	0.1724	0.0000	0.0000
al_00023	1.0000	1.0000	1.0000	0.1724	0.0000	0.0000
Similaridade:	0.996154	(99.6154%)				
Variáveis	compila	executa	nota	article_berry_blank_lines	article_berry_comments	article_berry_constant_definitions
al_00009	1.0000	1.0000	1.0000	0.1724	0.0000	0.0000
al_00014	1.0000	1.0000	1.0000	0.5517	0.0000	0.0000
Similaridade:	0.98235	(98.235%)				

Figura 5. Análise de plágios

O *PCodigo II* apresenta-se, portanto, como uma ferramenta muito útil para avaliação diagnóstica da aprendizagem de programação e que muito pode contribuir para professores entenderem como seus alunos programam e por que eles têm dificuldades e, a partir dessa compreensão, reorientar o ensino de forma a promover êxitos coletivos de aprendizagem.

5 | CONSIDERAÇÕES FINAIS

Este trabalho, que também foi publicado no Simpósio Brasileiro de Informática na Educação 2017 (NEVES, 2017), apresentou o sistema *PCodigo II* como um instrumento de análise da aprendizagem de programação a partir de códigos-fontes por mapeamento de perfis em 348 métricas de software. Os resultados da primeira experiência apontam o *PCodigo II* como uma ferramenta adequada para análise multidimensional da aprendizagem de programação.

Como trabalhos futuros a partir deste, sugerimos selecionar automaticamente as métricas mais relevantes para a avaliação de diferentes exercícios de programação e criar significados de grupos de métricas.

Em resumo, o *PCodigo II* apresenta-se como uma ferramenta de avaliação diagnóstica eficaz que auxilia professores a melhor compreenderem os processos de aprendizagem de seus alunos. Dessa forma, a contribuição dessa ferramenta para favorecer a aprendizagem de programação é ter um mecanismo para mostrar como os alunos programam, apontar dificuldades individuais e comuns em um turma e também reconhecer boas práticas de programação que caracterizam programadores hábeis e competentes.

REFERÊNCIAS

BAEZA-YATES, Ricardo et al. **Modern information retrieval**. New York: ACM press, 1999. Berry, R. and Meekings, B. A. (1985). A style analysis of C programs. *Communications of the ACM*, 28(1):80–88.

CURTIS, Bill et al. **Measuring the psychological complexity of software maintenance tasks with the Halstead and McCabe metrics**. *IEEE Transactions on software engineering*, n. 2, p. 96-104, 1979.

KARYPIS, George. **CLUTO-a clustering toolkit**. Minnesota Univ Minneapolis Dept of Computer Science, 2002.

KOLDE, Raivo. **Pheatmap: pretty heatmaps**. R package version, v. 61, 2012.

MUNSON, Jonathan P. **Metrics for timely assessment of novice programmers**. *Journal of Computing Sciences in Colleges*, v. 32, n. 3, p. 136-148, 2017.

NEVES, Adler et al. **Mapeamento Automático de Perfis de Estudantes em Métricas de Software para Análise de Aprendizagem de Programação**. In: Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE). 2017. p. 1337.

OLIVEIRA, Márcia Gonçalves; CIARELLI, Patrick Marques; OLIVEIRA, Elias. **Recommendation of programming activities by multi-label classification for a formative assessment of students**. *Expert Systems with Applications*, v. 40, n. 16, p. 6641-6651, 2013.

OLIVEIRA, M.; NOGUEIRA, M.; OLIVEIRA, Elias. **Sistema de apoio à prática assistida de programação por execução em massa e análise de programas**. In: XIV Workshop de Educação em Computação (WEI)-SBC 2015. 2015.

PETTIT, Raymond et al. **An empirical study of iterative improvement in programming assignments**. In: Proceedings of the 46th ACM Technical Symposium on Computer Science Education. ACM, 2015. p. 410-415.

PIETERSE, Vreda. **Automated assessment of programming assignments**. In: Proceedings of the 3rd Computer Science Education Research Conference on Computer Science Education Research. Open Universiteit, Heerlen, 2013. p. 45-56.

CLASSIFICAÇÃO DE CÓDIGOS C USANDO MEDIDAS DE SIMILARIDADE PARA APOIO AO ENSINO DE PROGRAMAÇÃO

José Carlos Campana Filho

Universidade Federal do Espírito Santo (Ufes)
Vitória - ES

Elias Silva de Oliveira

Universidade Federal do Espírito Santo (Ufes)
Vitória - ES

Márcia Gonçalves de Oliveira

Instituto Federal do Espírito Santo (Ifes)
Vitória – ES

RESUMO O aumento de cursos de programação de computadores em plataformas de ensino a distância tem gerado uma grande quantidade de questões e códigos. No entanto, essa base de conhecimento nem sempre está organizada de forma adequada para ser reaproveitada. Com o objetivo de auxiliar professores na geração dessa base, propomos uma melhoria em uma abordagem de classificação de códigos em Linguagem C baseada em medidas de similaridade para agrupar códigos de programação por tema ou por tipo de problema. A contribuição desse processo de classificação é a geração de uma base de questões com códigos de soluções associados, que pode ser utilizada como fonte de pesquisa ou para correção automática de questões de programação.

PALAVRAS-CHAVE classificação automática, medidas de similaridade, programação.

ABSTRACT The increase in computer programming courses in distance learning platforms has generated a lot of questions and codes. This knowledge base is not always organized in a suitable form to be reused. In order to assist teachers in the generation of the database by grouping programming codes by theme, or kind, of the problem, we propose an improvement in the classification of codes approach for the C language based on similarity measures. The contribution of this classification process is the generation of a base of questions associated to the code solutions that can be used as a source for research and for automatic assessment of program issues.

KEYWORDS automatic classification, similarity measures, programming.

1 | INTRODUÇÃO

A motivação deste artigo surgiu a partir de um problema que professores do Programa de Educação Tutorial (PET) da Universidade Federal do Espírito Santo (Ufes) enfrentaram (VALENTIM *et al.* 2014).

O PET/Ufes possui um histórico de atividades de programação cujas soluções foram desenvolvidas em programas escritos em Linguagem C e submetidos por seus alunos

na plataforma *Moodle*. Todavia, os enunciados das atividades foram desvinculados de seus códigos ao longo do tempo. Dessa forma, uma vez que esses códigos ficaram sem associações com seus enunciados, neste trabalho, temos como objetivo reconstruir essas associações.

Nossa proposta de solução é, portanto, a utilização de um processo de classificação desses códigos em que os enunciados passam a representar as classes almejadas. Essa estratégia visa apoiar essa tarefa de reconstrução de uma base de dados de cerca de 100 atividades e mais de 3000 códigos que foram submetidos ao longo do programa do PET/Ufes.

Essa base de dados gerada pode ser usada como gabarito de questões, servindo de referência para professores em uma correção manual, ou ser usada como conjunto de treino para tecnologias desenvolvidas que realizam correção automática de questões de programação (MOREIRA E FAVERO, 2009) e para sistema de apoio à prática assistida de programação por execução em massa e análise de programas (OLIVEIRA *et al.* 2015)

Para classificarmos os códigos, usamos como solução a abordagem de classificação proposta por Baby *et al.* (2014), que classifica códigos escritos em Linguagem C que tratam a mesma classe de atividade, isto é, agrupa códigos que tratam um mesmo tema (estruturas de árvore, fila, matriz e outros temas).

Assume-se, dessa forma, que os códigos podem ser classificados de acordo com a ocorrência de determinados termos, sejam eles identificadores, palavras-chave, operadores ou símbolos. Os códigos são decompostos em vetores de termos e a classificação é baseada no cálculo de similaridades dos vetores dado por um conjunto de 36 métricas de distância.

Como contribuição de melhoria no processo proposto por Baby *et al.* (2014), foi realizada uma análise sintática e semântica (CHA, 2007) do conjunto das 36 medidas de similaridade, visando reduzir esse conjunto e melhorar o desempenho do processamento.

Este trabalho está organizado conforme a ordem a seguir. Na Seção 2, apresentamos os trabalhos relacionados a nossa proposta. Na Seção 3, explicamos a metodologia utilizada para classificar os códigos. Na Seção 4, apresentamos a análise feita nas métricas de similaridade em busca de melhoria de desempenho. Na Seção 5, relatamos os experimentos. Na Seção 6, concluímos com as considerações finais e trabalhos futuros.

2 | TRABALHOS RELACIONADOS

Muitos trabalhos já foram realizados em relação à similaridade de códigos de programação, seja para detecção de plágio ou mesmo para classificação. Muitas

ferramentas estão disponíveis para realizar a análise de similaridade entre códigos, entre as quais destacamos o *JPlag* (PRECHELT *et al.*, 2002), *MOSS* (AIKEN, 2014) e *Sherlock* (PIKE, 2012). A proposta desenvolvida neste artigo se difere desses trabalhos, principalmente no tocante ao cálculo de distância entre os arquivos para determinar a similaridade. Enquanto nos trabalhos relacionados o grau de similaridade é dado pelo cálculo de distância de uma medida (*cosseño* ou euclidiana, normalmente), neste artigo ele é dado pelo cálculo de distâncias de um conjunto de medidas.

O *JPlag* é uma ferramenta que reconhece semelhanças entre conjuntos de arquivos de códigos-fontes. O *JPlag* considera a sintaxe da linguagem de programação e a estrutura de programas ao compará-los, o que é bem adequado para detectar tentativas de disfarces de semelhanças entre os arquivos plagiados. O *JPlag* suporta códigos em Java, C#, C, C++, *Scheme* e texto em linguagem natural. O *JPlag* já desempenhou importante papel em casos de propriedade intelectual, onde ele tem sido usado com sucesso por peritos.

O Measure of Software Similarity (*MOSS*) é um sistema automático para a análise de similaridade de programas. A principal aplicação do *MOSS* foi na detecção de plágios realizados em aulas de programação e, desde o seu desenvolvimento em 1994, o *MOSS* tem sido muito eficaz nesse papel. O *MOSS* é capaz de analisar similaridades entre códigos escritos em várias linguagens de programação, dentre as quais destacamos C, C++, Java, C#, Python, Visual Basic, Javascript, Fortran, Haskell, Lisp, Pascal, Ada e Perl.

O *Sherlock*, por sua vez, é um programa que verifica similaridades entre documentos de textos. Ele gera uma assinatura digital (*hash*) para uma palavra ou sequência de palavras e compara para detectar semelhanças.

A assinatura digital é um número que representa um conjunto de palavras convertidas em uma série de bits. O *Sherlock* é implementado em C e possui código aberto, podendo ser aprimorado pela comunidade. Como essa ferramenta compara códigos como documentos de texto, é interessante que se faça uma preparação do arquivo do código antes de submetê-lo ao *Sherlock*.

Um trabalho muito interessante é o de Baby *et al.* (2014), que trata a análise de similaridade um pouco diferente ao determinar a similaridade entre arquivos por um conjunto de medidas de distância. Por ser um tópico no estudo de similaridade ainda não muito explorado, o nosso trabalho baseou-se nessa proposta, que será apresentada em mais detalhes na próxima seção.

Considerando a proposta de Baby *et al.* (2014), visando uma performance melhor, usamos os conceitos de Cha (2007) para reduzir o conjunto de medidas de similaridade utilizadas. Na proposta de Cha (2007) é realizada uma análise sintática e semântica de várias medidas de similaridade e elas são agrupadas conforme algumas características. A melhoria nesse processo irá, conseqüentemente, contribuir para a geração de uma base de enunciados e códigos de programação.

3 | METODOLOGIA

Nesta seção descrevemos o método usado neste artigo, que é baseado no método proposto por Baby *et al.* (2014). Na Figura 1, é ilustrado o processo desse método, onde os dados são primeiramente divididos em treino e teste e o modelo é então avaliado comparando as predições com o conjunto de teste.

De acordo com a Figura 1, uma lista de *tokens* é criada contendo as funções nativas e operadores usados na Linguagem C, sendo que cada *token* é associado a um código inteiro único.

Na conversão, o código em Linguagem C é transformado em um vetor de termos, onde cada dimensão é a frequência de ocorrência de um termo, que pode ser uma palavra-chave, um operador, um literal, uma constante ou um outro símbolo. Assim, cada termo é convertido no valor inteiro equivalente e representa uma posição no vetor de frequência. O valor na posição referente ao termo é, portanto, a sua frequência de ocorrência em um código-fonte.

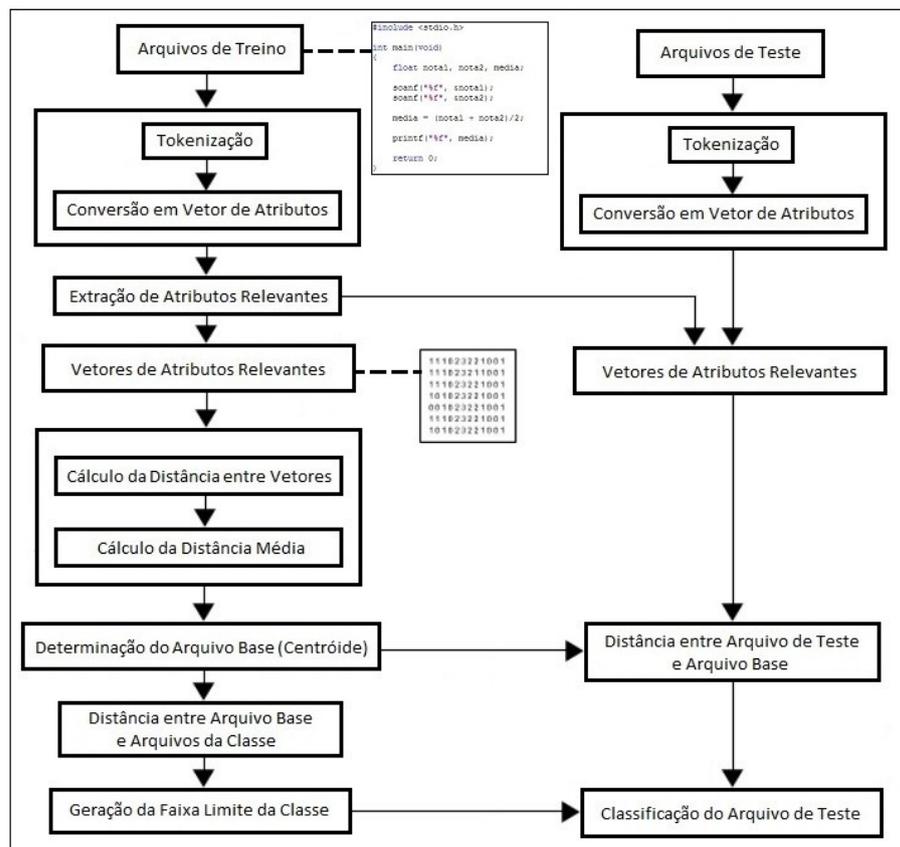


Figura 1. O método proposto

A Tabela 1 mostra um exemplo simples de conversão de código C em um vetor V de frequência dos termos. Nesse exemplo, os valores do vetor V correspondem à

frequência dos termos *int*, *main*, "{", *literal (x e y)*, "=", *constante (10 e 0)*, ";", *return* e "}", respectivamente. A similaridade entre dois códigos é obtida através do cálculo da distância entre dois vetores, onde P_i é o *i-ésimo* termo do primeiro vetor e Q_i é o *i-ésimo* termo do segundo vetor.

Nem todo termo é necessário para representar as características de uma classe. Somente os termos presentes em pelo menos 40% dos arquivos da classe são considerados relevantes para representar a classe, os outros são descartados. Em uma classe que contém 26 programas, por exemplo, termos presentes em mais de 40% dos 26 arquivos (ou seja, presentes em mais de dez arquivos) são selecionados como atributos relevantes.

A distância entre os vetores que representam os arquivos de uma classe é calculada para encontrar a similaridade entre esses arquivos. Dessa forma, quanto menor a distância, mais similares são dois arquivos.

Para verificar a similaridade entre os arquivos, utilizamos 36 medidas de distância, que são apresentadas na Tabela 2.

Para cada medida, é gerada uma matriz de distâncias $n \times n$, onde n é o número de programas reunidos em uma classe. Em seguida, é gerada outra matriz $n \times 36$ com as médias de distância para cada arquivo e medida.

A partir dessa matriz, é selecionado um arquivo-base, que é o centróide da classe, para cada medida. O centróide é o arquivo com a menor distância em determinada medida. O centróide pode ser ou não o mesmo arquivo para cada uma das 36 medidas de distância.

Para gerar a faixa limite da classe para cada medida de distância, é calculada a distância entre os arquivos-base de cada medida de distância e os arquivos similares. Se o número de arquivos que possuem valor de distância menor que a distância média é maior que o número de arquivos com distância maior que a distância média, então a faixa limite é a menor distância até a média, caso contrário, a faixa é a distância média até a maior distância.

Código	Tokens	Posição do token no vetor	Vetor de frequências
int main { int x = 10; int y = x; return 0; }	int	1	$v = \{3, 1, 1, 3, 2, 2, 3, 1, 1\}$
	main	2	
	{	3	
	literal (x e y)	4	
	=	5	
	constante (10 e 0)	6	
	;	7	
	return	8	
	}	9	

Tabela 1. Exemplo de conversão de código em vetor de frequências de termos

Para classificar arquivos de teste, é avaliada a distância entre o centróide e o

teste, e se o valor está dentro da faixa limite em mais de 50% das medidas, o arquivo é dito como pertencente à classe.

<i>Additive Symmetric</i>	<i>Average</i>	<i>Bhattacharrya</i>	<i>Canberra</i>
<i>Chebyshev</i>	<i>City Block</i>	<i>Clark</i>	<i>Czekanowski</i>
<i>Dice</i>	<i>Divergence</i>	<i>Euclidean</i>	<i>Gower</i>
<i>Harmonic Mean</i>	<i>Hellinger</i>	<i>Intersection</i>	<i>Jeffreys</i>
<i>Jensen Difference</i>	<i>Jensen Shannon</i>	<i>Kdivergence</i>	<i>Kulczynski</i>
<i>Kullbackliebler</i>	<i>Kumar Johnson</i>	<i>Lorentzian</i>	<i>Matusita</i>
<i>Neyman</i>	<i>Probabilistic Symmetric</i>	<i>Ruzicka</i>	<i>Soergel</i>
<i>Sorensen</i>	<i>Squared</i>	<i>Squared Euclidean</i>	<i>Squared Chord</i>
<i>Taneja</i>	<i>Tanimoto</i>	<i>Topsoe</i>	<i>Wave Hedges</i>

Tabela 2. Conjunto das 36 medidas de similaridade da abordagem original

4 | ANÁLISE DO CONJUNTO DE MÉTRICAS

Para tentar melhorar o desempenho na classificação, focamos em alterar o conjunto de medidas propostas por Baby et al. (2014). Para as fórmulas apresentadas nesta seção, considere d o número de termos do vetor, P_i o i -ésimo termo do primeiro vetor e Q_i o i -ésimo termo do segundo vetor.

Observando as 36 métricas utilizadas, e considerando as análises semânticas e sintáticas de Cha (2007), pode-se perceber que algumas métricas são equivalentes, ou seja, produzem resultados idênticos, como Soergel e Tanimoto, nas equações (1) e (2).

$$d_{Kdivergence} = \frac{\sum_{i=1}^d |P_i - Q_i|}{\sum_{i=1}^d \max(P_i, Q_i)} \quad (1)$$

$$d_{Topsoe} = \frac{\sum_{i=1}^d \max(P_i, Q_i) - \sum_{i=1}^d \min(P_i, Q_i)}{\sum_{i=1}^d \max(P_i, Q_i)} \quad (2)$$

Outras métricas são proporcionais, como *City Block* e *Gower*, em (3) e (4).

$$d_{CityBlock} = \sum_{i=1}^d |P_i - Q_i| \quad (3)$$

$$d_{Gower} = \frac{1}{d} \sum_{i=1}^d |P_i - Q_i|, \quad d_{Gower} = \frac{1}{d} d_{CityBlock} \quad (4)$$

Ainda temos métricas que são assimétricas, ou seja, a distância entre P e Q , $d(P, Q)$ é diferente de $d(Q, P)$, como a métrica $Kdivergence$, na equação (5). A métrica $Topsoe$ é uma versão simétrica da $Kdivergence$, na equação (6).

$$d_{Kdivergence} = \sum_{i=1}^d P_i \ln \frac{2P_i}{P_i + Q_i} \quad (5)$$

$$d_{Topsoe} = \sum_{i=1}^d \left[P_i \ln \left(\frac{2P_i}{P_i + Q_i} \right) + Q_i \ln \left(\frac{2Q_i}{P_i + Q_i} \right) \right] \quad (6)$$

As métricas idênticas ou proporcionais não trazem novas informações sobre a similaridade de arquivos. Portanto, foram escolhidas apenas uma de cada métrica idêntica ou proporcional. As métricas assimétricas foram descartadas, mantendo apenas as suas versões simétricas.

Com isso o número de métricas passou de 36 para 26. Foi adicionada a este novo conjunto a métrica *coseno*, pois é bastante mencionada na literatura (BAEZA e RIBEIRO, 1999) e não fazia parte do conjunto inicial. O novo conjunto passou a ter então 27 métricas, conforme a Tabela 3.

<i>Additive Symmetric</i>	<i>Average</i>	<i>Bhattacharrya</i>	<i>Canberra</i>
<i>Chebyshev</i>	<i>City Block</i>	<i>Clark</i>	-
<i>Dice</i>	<i>Divergence</i>	<i>Euclidean</i>	-
<i>Harmonic Mean</i>	-	<i>Intersection</i>	<i>Jeffreys</i>
<i>Jensen Difference</i>	<i>Jensen Shannon</i>	-	<i>Kulczynski</i>
-	<i>Kumar Johnson</i>	<i>Lorentzian</i>	<i>Matusita</i>
-	-	<i>Ruzicka</i>	-
-	<i>Squared</i>	<i>Squared Euclidean</i>	<i>Squared Chord</i>
<i>Taneja</i>	<i>Tanimoto</i>	-	<i>Wave Hedges</i>
<i>Cosseno</i>			

Tabela 3. Novo conjunto com 27 medidas de similaridade

5 | EXPERIMENTOS E RESULTADOS

Para o experimento deste trabalho, foram utilizadas duas bases de dados: a base utilizada no trabalho de Baby *et al.* (2014) e uma base de uma turma de programação formada por alunos do PET/UFES. O desempenho de classificação foi analisado com a métrica *Acurácia*.

Para a validação do modelo foi usada a técnica de validação cruzada com o método o *leave-one-out*. O método *leave-one-out* consiste em retirar um elemento da base para teste e utilizar o restante da base como conjunto de treino. Esse processo é repetido para todos os elementos. No final das iterações, calcula-se a *Acurácia* sobre as predições.

a) Base do artigo de referência

A base utilizada por Baby *et al.* (2014) está dividida em quatro classes: *Btree*, *Crque*, *Dqueue* e *Matrix*, que agrupam códigos referentes aos temas *árvore B*, *fila circular*, *fila duplamente encadeada* e *matriz*, respectivamente. Cada uma dessas classes reúne 26 arquivos.

O processo de classificação foi realizado utilizando o conjunto inicial de medidas e, depois, o novo conjunto. A Tabela 4 mostra os resultados obtidos. As predições obtidas para as classes *Btree* e *Matrix* foram muito boas. Já as classes *Crque* e *Dqueue* não obtiveram a mesma qualidade de predição. Para todas as classes, alguns arquivos passaram a ser classificados na classe correta somente quando executado com as 27 medidas, o que sugere uma melhora na classificação.

Classe	Total de arquivos	Qtde acertos	Qtde acertos	% de acerto	% de acerto
		36 medidas	27 medidas	36 medidas	27 medidas
Btree	26	22	23	84.61	88.46
Crque	26	11	11	42.30	42.30
Dqueue	26	11	14	42.30	53.84
Matrix	26	21	22	80.77	84.61
Total	104		Média	62.50	67.30

Tabela 4. Acurácia obtida com conjuntos de 36 e 27 medidas de similaridade para a base do artigo de referência

b) Base de Exercícios do PET/UFES

A base de exercícios do PET/UFES utilizada foi obtida de uma turma de alunos de Introdução à Programação em Linguagem C do projeto *Introcomp* (MENESES *et al.*, 2015). Essa base possui treze classes, que são atividades, cada uma com quantidade de arquivos variados em um total de 591 arquivos.

A Tabela 5 mostra a quantidade de arquivos classificados corretamente para os dois conjuntos de medidas da base do PET/UFES.

Ao executar o processo de classificação, verificamos que alguns arquivos passaram a ser classificados na classe correta somente quando executado com as 27 medidas, como havia ocorrido na outra base.

Alguns arquivos foram classificados em mais de uma classe. Isso era esperado e se deve ao fato de algumas atividades abordarem conceitos parecidos de programação.

Nesse caso, alunos submeteram códigos de programas com conjunto de *tokens* parecidos. Situações como essa devem ser tratadas com a supervisão do professor para a indicar a correta classificação.

Classe	Total de arquivos	Qtde acertos	Qtde acertos	% de acerto	% de acerto
		36 medidas	27 medidas	36 medidas	27 medidas
Questão 1	22	18	18	81.82	81.82
Questão 2	23	20	19	86.96	82.61
Questão 3	26	21	22	80.77	84.62
Questão 4	31	14	18	45.16	58.06
Questão 5	50	42	44	84.00	88.00
Questão 6	50	38	39	76.00	78.00
Questão 7	52	34	36	65.38	69.23
Questão 8	53	36	35	67.92	66.04
Questão 9	53	40	44	75.47	83.02
Questão 10	56	20	20	35.71	35.71
Questão 11	58	47	47	81.03	81.03
Questão 12	58	39	45	67.24	77.59
Questão 13	59	51	51	86.44	86.44
Total	591		Média	71.84	74.78

Tabela 5. Acurácia obtida com conjuntos de 36 e 27 medidas de similaridade para a base do PET/UFES

6 | CONSIDERAÇÕES FINAIS

Considerando o desempenho de classificação dos códigos C dos dois conjuntos de métricas, observamos que o novo conjunto obteve uma acurácia média de arquivos classificados corretamente melhor do que o primeiro, para as duas bases utilizadas, isto é, 67.30% contra 62.50% na primeira base, e 74.78% contra 71.84% na base do PET. Com isso, conclui-se que, com a retirada das medidas idênticas, proporcionais e assimétricas, e a inclusão da medida de similaridade *cosseño*, evidenciamos melhor a similaridade entre os códigos, e melhoramos o desempenho do processo.

Com esses resultados, o processo proposto se mostra capaz de solucionar o problema tratado nesse artigo, que é gerar uma base com uma grande quantidade de códigos associados a enunciados de questões e disponibilizar para professores do PET, ou para outros professores que tenham o mesmo problema. Essa base pode ser usada, por exemplo, como um repositório de pesquisa para alunos e professores ou como base de treino para a realização de correções automáticas.

Como trabalho futuro a partir deste, temos a análise de outras medidas e outros

pontos que podem ser alterados na abordagem inicial, como a *tokenização*, extração de atributos relevantes e a faixa de corte das classes. Também como trabalho futuro, pode-se aplicar essa abordagem de similaridade para detecção de possíveis plágios de códigos de programação.

REFERÊNCIAS

BABY, Julie et al. **Distance indices for the detection of similarity in C programs**. In: Computation of Power, Energy, Information and Communication (ICCPEIC), 2014 International Conference on. IEEE, 2014. p. 462-467.

BAEZA-YATES, Ricardo et al. **Modern information retrieval**. New York: ACM press, 1999.

CHA, Sung-Hyuk. **Comprehensive survey on distance/similarity measures between probability density functions**. City, v. 1, n. 2, p. 1, 2007.

MENESES, Leonardo F. et al. **IntroComp: Atraindo alunos do ensino médio para uma instigante experiência com a programação**. In: Anais do XXIII Workshop sobre Educação em Computação (WEI 2015), Recife, PE, 2015.

MOREIRA, Mireille Pinheiro; FAVERO, Eloi Luiz. **Um ambiente para ensino de programação com feedback automático de exercícios**. In: Workshop sobre Educação em Computação (WEI 2009). 2009.

AIKEN, Alex. **Moss (measure of software similarity) plagiarism detection system**. Disponível em: <http://www.cs.berkeley.edu/moss/>, 2000. Acesso em 16/04/2018.

OLIVEIRA, M.; NOGUEIRA, M.; OLIVEIRA, Elias. **Sistema de apoio à prática assistida de programação por execução em massa e análise de programas**. In: XIV Workshop de Educação em Computação (WEI-SBC), 2015.

PIKE, R. **The sherlock plagiarism detector**. Disponível em: <http://sydney.edu.au/engineering/it/~scilect/sherlock/>. Acesso em 16/04/2018, v. 21, 2012.

PRECHELT, Lutz; MALPOHL, Guido; PHILIPPSEN, Michael. **Finding plagiarisms among a set of programs with JPlag**. J. UCS, v. 8, n. 11, p. 1016, 2002.

VALENTIM, R. et al. **Em busca de uma metodologia para a disseminação em massa do ensino de programação**. Seminário Nacional de Inclusão Digital (SENID), Passo Fundo, RS, SBC, 2014.

EIXO 2 – JOGOS DIGITAIS E GAMIFICAÇÃO

APRESENTAÇÃO

O segundo eixo entra em jogos digitais e gamificação, no primeiro trabalho focado na aprendizagem de pessoas com deficiência visual, possuem uma diferente percepção devido às suas particularidades tenta garantir no processo pedagógico uma inclusão digital e social destas pessoas. No próximo artigo o tema de jogos construcionistas foi inserido no processo de aprendizagem e utilizado na construção de quatro jogos educativos, simplificando a complexidade e direcionando o processo de criação de jogos.

A abordagem gamificação ajuda a resolver problemas motivacionais dos alunos neste último artigo deste eixo, promovendo um engajamento no processo de aprendizagem, fazendo uma revisão sistemática da literatura para verificar o que ocorre em ambientes educacionais gamificados.

Everson Mario Novak

Mestrando em Informática - PUCPR

BEM EXPRESSÕES: JOGO DIGITAL VOLTADO PARA O ENSINO INCLUSIVO DA MATEMÁTICA

André Luis Bitencourt Fernandes

Universidade Estadual de Feira de Santana,
Departamento de ciências exatas
Feira de Santana – Bahia

Claudia Pinto Pereira

Universidade Estadual de Feira de Santana,
Departamento de ciências exatas.
Feira de Santana – Bahia

Kayo Costa de Santana

Universidade Estadual de Feira de Santana,
Departamento de ciências exatas
Feira de Santana – Bahia

Ana Jaize de Oliveira Silva Santos

Universidade Estadual de Feira de Santana,
Departamento de ciências exatas
Feira de Santana – Bahia

Bruno Gonzaga de Mattos Vogel

Universidade Estadual de Feira de Santana,
Departamento de ciências exatas
Feira de Santana – Bahia

RESUMO O uso das tecnologias vem gradativamente sendo integrado ao processo educacional. Vídeos, imagens, jogos analógicos e digitais são introduzidos na rotina dos estudantes e professores, a fim de garantir uma melhor efetividade no processo de aprendizagem. Pessoas com deficiência, mesmo envoltos em uma mesma realidade, possuem uma diferente percepção devido às

suas particularidades. Por esse motivo, torna-se cada vez mais importante aliar ao processo educacional o uso de tecnologias, tentando garantir uma universalidade no processo pedagógico, que promova a inclusão digital e social destas pessoas. Neste contexto, encontra-se o BEM (*Blinds, Education and Mathematics*), uma ferramenta digital focada para a inclusão da pessoa com deficiência visual e direcionada para o ensino matemática, cuja evolução e atual estado da obra serão destacados neste artigo.

PALAVRAS-CHAVE Tecnologia de Informação e Comunicação; Inclusão; Matemática; Objeto de Aprendizagem.

ABSTRACT Technologies are increasingly being used on educational process. Tools such as videos, images, analogue and digital games are being introduced on teaching process to improve the learning effectiveness.

People with impairments, independent of they social context, have different perceptions due to they particular needs, exposing the important necessity of aligning the usage of technologies with the educational process to ensure a universality on pedagogic process. In this context, BEM (Blinds, Education and Mathematics) is a digital tool which aims to include people with visual impairment and teach them math. BEM's evolution and current state of work are highlighted throughout this paper.

KEYWORDS: Information and communication technologies; Inclusion; Mathematics; Learning objects.

1 | INTRODUÇÃO

Com o avanço das tecnologias digitais no último século, é perceptível a mudança no cotidiano das pessoas (PINHEIRO; SILVEIRA e BAZZO, 2007). Atualmente, com tantos recursos oferecidos, a imersão do indivíduo em meio aos dispositivos eletrônicos molda a realização de suas atividades em vários âmbitos de sua vida. Com o uso de um *smartphone*, por exemplo, é possível a realização de várias tarefas corriqueiras, tais como a realização de contas (i.e. uso de calculadoras), entretenimento (e.g. vídeos, jogos, notícias e etc.) e, até mesmo, atividades mais complexas, como o pagamento de boletos e a transferência de valores (através de aplicativos de contas bancárias).

Modernidade e tecnologias digitais acabam se entrelaçando e se tornando “sinônimos”, de tal modo que estas tecnologias agem como modificadores da sociedade, impulsionando avanços que impactam e influenciam a realidade. De maneira semelhante, o processo pedagógico não se distancia dessa realidade. Através das Tecnologias de Informação e Comunicação (TIC), é possível oferecer ao educando o desenvolvimento de habilidades motoras e cognitivas, proporcionar a dinamização do processo educacional, ampliar a ludicidade do processo de ensino aprendizagem e instigar, gradativamente, uma exploração positiva dos recursos oferecidos pelas TIC (OLIVEIRA; LUDWIG e FINCO, 2011).

Segundo Pinto (1993, p. 45), “a forma da educação é função de seus fins sociais. Tem que ser em cada caso aquela que se adapta ao conteúdo, isto é, à condição do educando, suas possibilidades imediatas de ascensão cultural”, desta forma, deve-se ficar atento para o fato de que o processo educacional não deve ser moldado apenas pelo conteúdo programático, mas também por todo o contexto social ali envolto. Não basta, portanto, trazer apenas tecnologias para dentro das salas de aula e esperar alterações do processo pedagógico, sem que sejam pensadas e planejadas novas estratégias metodológicas. Para que sejam percebidas alterações significativas, o processo educacional deve ser reestruturado e adaptado para lidar com os interesses e as particularidades dos educandos, e as diferentes formas de aprender. Em outras palavras, não basta apenas utilizar as TIC para reproduzir o que estava sendo feito, é importante que sejam utilizadas como ferramentas auxiliares, que possam atender as condições de ensino e de aprendizagem de diferentes indivíduos.

Uma das formas possíveis de facilitar o processo de ensino é através da ludicidade proporcionada pelos objetos de aprendizagem (OA), que, são recursos digitais ou não digitais, utilizados para apoiar a aprendizagem. Alguns exemplos de objetos digitais de aprendizagem são: vídeos, imagens e, até mesmo, jogos educativos. Outro recurso

importante utilizado para facilitar o ensino e a mobilidade de diversos grupos sociais são as tecnologias assistivas, que contribuem para a (in)formação de pessoas com deficiência, promovendo, desta maneira, maior autonomia/independência e inclusão sócio digital.

Envolvido neste contexto, membros do Grupo de Informática Conhecimento e Educação (GICE) da Universidade Estadual de Feira de Santana (UEFS) vem desenvolvendo, ao longo de quatro anos, ferramentas e jogos digitais que auxiliam no ensino-aprendizagem, com um enfoque para o público com deficiência visual (DV), dentre os quais se destacam os objetos de aprendizagem: *Blinds*, *Basic Education* (BBE), jogo da velha digital com utilização de *quizz*, que permite a prática/ensino de diferentes componentes curriculares (SANTANA et al., 2017) e o *Blinds, Education and Mathematics* (BEM), utilizado para o ensino lúdico de operações matemáticas (MACEDO et al., 2017).

O BEM é um jogo digital em forma de tabuleiro, criado para o ensino-aprendizagem da matemática, inicialmente pensado apenas para a realização das operações matemáticas básicas (e.g. soma, subtração, multiplicação e divisão). Entretanto, ele evoluiu, ao longo do tempo, para expressões aritméticas, sempre pensado para a inclusão de pessoas com deficiência visual. Este artigo objetiva, portanto, apresentar a evolução deste jogo (BEM), e, em especial, detalhar sua nova versão *web*, para o ensino de expressões aritméticas.

Neste sentido, o artigo se divide em 5 seções, além desta introdução. A seção 2 traz o uso das TIC na perspectiva da educação inclusiva; a seção 3 apresenta os conceitos de tecnologias assistivas e acessibilidade, visto que são conceitos associados à proposta do jogo BEM; as seções 4 e 5 apresentam, não só a evolução da ferramenta, em suas primeiras versões, como principalmente a nova versão, o BEM Expressões Aritméticas, refletindo o seu uso por estudantes e professores para aprender e ensinar a matemática para pessoas com deficiência visual, proporcionando a inclusão sócio digital desse público; e, por fim, na seção 6, são apresentadas as considerações finais sobre a ferramenta bem como possíveis melhorias para versões futuras.

2 | A EDUCAÇÃO INCLUSIVA E O USO DAS TIC

Para oferecer uma educação de qualidade aos estudantes com necessidades especiais, a educação inclusiva vem possibilitando que eles caminhem, lado a lado, em sua trajetória escolar, junto com seus colegas sem deficiência. Essa aproximação pode, não só promover melhorias nas práticas de ensino, como também ampliar o desempenho escolar destes estudantes (HEHIR et al, 2016). Os professores, neste contexto, precisam refletir suas práticas pedagógicas e os estudantes ditos “normais”

precisam se sensibilizar e compreender que todos são diferentes e o quanto é importante conviver e respeitar essas diferenças, dentro e fora do espaço escolar.

Segundo Silva (2009), se existe uma classe homogênea, na qual os estudantes realizam as mesmas tarefas, este espaço terá de ceder lugar à classe heterogênea, na qual pessoas com diferenças convivem e aprendem de acordo com suas particularidades. O autor acredita que, neste momento, existe a necessidade de se pensar em um novo modelo educacional, a fim de incluir estes estudantes com necessidades específicas. Vale lembrar, também, que todos nós somos diferentes uns dos outros, ainda que não apresentemos deficiências conhecidas e/ou categorizadas.

Nessa perspectiva, o uso das TIC nos ambientes escolares vem contribuindo, em especial, para obtenção do desenvolvimento educacional, exigindo que a educação se renove, tornando o ensino mais criativo, estimulando o interesse pela aprendizagem e trazendo inovações e facilidades para os ambientes acadêmicos. Giroto e outros (2012, p.7) apontam que:

As Tecnologias de Informação e Comunicação (TICs) apresentam-se como promissoras para a implementação e consolidação de um sistema educacional inclusivo, pelas suas possibilidades inesgotáveis de construção de recursos que facilitam o acesso às informações, conteúdos curriculares e conhecimentos em geral, por parte de toda a diversidade de pessoas dentre elas as que apresentam necessidades especiais.

Dessa forma, a utilização das tecnologias de informação e comunicação como meio facilitador do processo de ensino-aprendizagem propõe melhorias no processo educacional e possibilita o aperfeiçoamento na transmissão e na aquisição do conhecimento, através das ferramentas e materiais digitais, tendo como principal objetivo favorecer a interatividade e também provocar mudanças nos métodos de ensino (SANTOS et al., 2016).

Portanto, ao utilizar a tecnologia como uma aliada do professor na educação inclusiva, podem-se ampliar as possibilidades de interação entre os estudantes, com ou sem deficiência, e entre estes e os *softwares* educativos e os objetos de aprendizagem. Entretanto, vale a pena lembrar que não adianta “tecnologia” sem mudança de “metodologia educacional” e sem intenção de mudança, para que a inclusão real desse público aconteça. É preciso que as tecnologias sejam usadas de maneira criativa, lúdica, com propósito pedagógico definido, para a geração de conhecimentos e para a (in)formação do cidadão, de tal modo que ele possa interagir dentro e fora do espaço escolar.

Como o jogo BEM é um objeto de aprendizagem voltado para a inclusão de pessoas com deficiência visual, ele também é considerado uma tecnologia assistiva, com recursos acessíveis. Dessa forma, é importante compreender esses dois conceitos, apresentados na seção seguinte.

3 | TECNOLOGIAS ASSISTIVAS E ACESSIBILIDADE

O controle do ambiente próprio, mobilidade, comunicação são, necessariamente, aspectos que envolvem a independência de cada ser humano. É nesta vertente que as Tecnologias Assistivas (TA) desenvolvem os seus avanços, habilitando funções desejadas pelo indivíduo, porém, impedidas por razões de deficiência ou envelhecimento (BERSCH, 2008). De forma introdutória ao conceito de TA, pode-se utilizar também uma das falas de Radabaugh (1993) citadas em Bersch (2008, p. 2), que diz: “para as pessoas sem deficiência a tecnologia torna as coisas fáceis. Para as pessoas com deficiência, a tecnologia torna as coisas possíveis”.

Quando se trata de instrumentos físicos, as TA possuem a sua representatividade, por exemplo, em cadeiras de rodas, andadores e adequadores posturais em carteiras de sala de aula (PELOSI et al., 2009). A abordagem de utilização de *software* como TA é mais recente, contudo segue a intenção de se tornar de tal forma familiar à sociedade, como são estes instrumentos físicos. Um grande passo para isto são as discussões que vêm ocorrendo, nas diretrizes nacionais de educação, sobre a construção de um projeto pedagógico heterogêneo pelos professores, com o uso de TA (PELOSI et al., 2009).

Neste contexto, os estudos de Rocha e Deliberato (2012) orientam que a utilização de TA em ambiente escolar necessita de estratégias específicas para cada tipo de ambiente e de estudante. Assim, informações dos profissionais, dos estudantes e do próprio ambiente são necessárias para a elaboração de recursos que atendam a estas especificações. Como as TA também se caracterizam como *softwares* que possibilitam a inclusão e a ampliação de habilidades, seu uso pode ser uma forma de quebrar barreiras, flexibilizar e facilitar o acesso e o manuseio de ferramentas digitais, adequadamente acessíveis (PASSERINO e MONTARDO, 2007).

É importante ressaltar também a diferença, proposta pelos mesmos autores, sobre acessibilidade e usabilidade, que podem ser facilmente confundidas. Enquanto a acessibilidade trata as condições de uso, sendo elas, como o usuário se apresenta frente a interfaces interativas, como deve funcionar esta troca e como se dará o acesso do usuário a informações, a usabilidade se propõe tratar as expectativas e a capacidade do usuário em perceber e entender as estratégias de utilização do *software*.

A ênfase maior para a construção destes tipos de TA digitais está na realização de estudos de fatores, normalmente, desconsiderados em sistemas de usuários sem deficiência. Indispensavelmente, deve existir, neste caso, a formação de cientistas, desenvolvedores de *software*, no âmbito multidisciplinar, envolvendo aspectos de educação; entendimento sobre as deficiências (e.g. visual, auditiva, motora) e jogos; psicólogos, dentre outros. Além disso, é importante lembrar que ambientes digitais de aprendizagem têm a possibilidade de serem inseridos em diversos grupos e espaços, talvez ainda não explorados, tais como salas de aula, clínicas e consultórios de terapia

de pessoas com deficiências e/ou transtornos, aulas de musicoterapia, salas de recursos, dentre outros.

4 | METODOLOGIA

O processo de desenvolvimento da ferramenta BEM, em todas as suas versões, obedeceu ao mesmo delineamento metodológico, como apresentado na Figura 1.

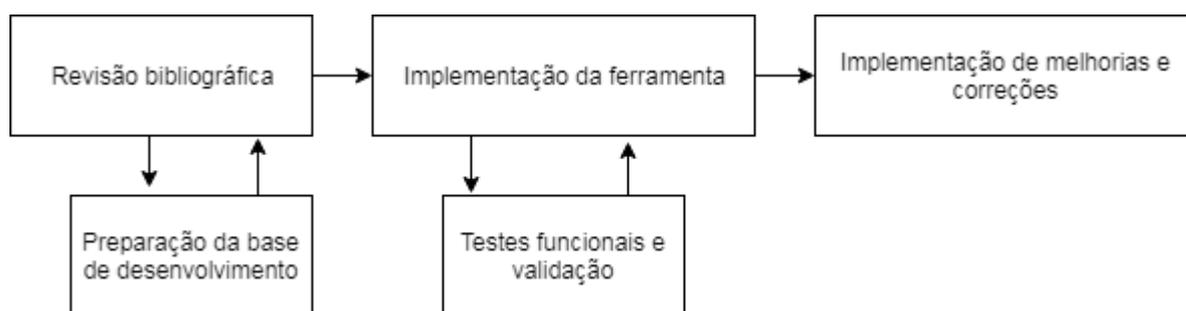


Figura 1 - Processo metodológico de construção das ferramentas

Fonte: Próprio autor (2018)

Inicialmente, foi feito um levantamento do referencial teórico, com o objetivo de se aproximar da temática que seria trabalhada. Foram feitas pesquisas e leituras sobre pessoas com deficiência visual, jogos educativos e a educação matemática. Além disso, foi feita também uma análise funcional da linguagem de programação e das bibliotecas que seriam utilizadas na ferramenta.

Em paralelo à etapa de revisão bibliográfica, foi feita a preparação do ambiente de desenvolvimento. Essa etapa compreendeu basicamente a instalação de todos os *softwares* necessários para a implementação da ferramenta. Em seguida, partiu-se para o desenvolvimento em si da ferramenta e, em paralelo, foram feitos testes funcionais para validação dos requisitos, na medida em que foram implementados.

Embora os procedimentos metodológicos tenham se mantido em todas as versões da ferramenta, a sua concepção, ao longo do período, foi se modificando, e, com ela, a necessidade de novos requisitos. Inicialmente, o BEM foi desenvolvido para abordar o conteúdo referente às operações básicas da matemática de forma isolada, com o diferencial de ser acessível a pessoas com deficiência visual, a partir de um motor de síntese de voz. A primeira versão da ferramenta foi desenvolvida utilizando a linguagem de programação Java, juntamente com o motor de síntese de voz da IBM (TECHTUDO, 2012). A escolha por estas tecnologias se deu por conta da facilidade de acesso às mesmas.

O requisito seguinte, adicionado em sua segunda versão, foi a implementação de um sistema de reconhecimento de voz, com o intuito de facilitar ainda mais a interação usuário/jogo e a acessibilidade para pessoas com deficiência visual. Com o reconhecimento e a síntese de voz implementados, a pessoa com DV poderia, além

da interação via teclado, também fazê-la através da voz, tanto como entrada, quanto como saída. Para tanto, foi necessário realizar a revisão bibliográfica acerca, não só do conceito e das possibilidades do reconhecimento de voz, como também sobre as tecnologias possíveis. Dentre as tecnologias encontradas, destacam-se: a JLapsApi (ALVES, 2015), o Sphinx-4 (CMUSPHINX, 2015), o Cloudgarden (CLOUDGARDEN, 2018) e o IBM Via Voice (TECHTUDO, 2012). Após análise destas tecnologias, decidiu-se pela utilização do IBM Via Voice como motor de reconhecimento de voz, pela questão de viabilidade e de tempo de projeto, uma vez que já era a mesma tecnologia utilizada na funcionalidade de síntese. O reconhecimento de voz trabalhado foi o de gramática pré-definida, com palavras e/ou frases preestabelecidas para acionamento das ações do *software*.

A primeira versão do BEM Expressões ampliou as funcionalidades do BEM, acrescentando a possibilidade de o usuário/estudante trabalhar também com expressões aritméticas. Entretanto, manteve a mesma interface e o uso das mesmas tecnologias de desenvolvimento (Java) e de motor de síntese de voz (IBM Via Voice), embora sem o recurso de reconhecimento.

Como o IBM Via Voice apresentava alguma dificuldade no momento de instalação, em especial considerando um usuário leigo, pensou-se em desenvolver a versão *web* do BEM Expressões, com dois objetivos básicos: evitar o processo de instalação, facilitando o acesso em qualquer navegador, e modificar o motor de síntese de voz, em busca de um gratuito, com voz menos “robotizada” e sem necessidade de instalação prévia. Este momento foi iniciado com a identificação de um novo motor de síntese de voz gratuito, que pudesse atender às necessidades dos usuários do ponto de vista da qualidade de síntese. Foi encontrado como tecnologia que atendesse a estes requisitos o *ResponsiveVoice* (RESPONSIVEVOICE, 2018) que se trata de uma biblioteca para *JavaScript* capaz de converter textos em áudio.

Como tecnologias utilizadas na implementação da interface visual da ferramenta foram utilizados o HTML5 e o CSS3, que foram escolhidas por serem tecnologias amplamente utilizadas para o desenvolvimento de aplicações *web*, e, desta forma, já contam com ampla documentação disponível, bem como o *JavaScript*, responsável pela dinâmica de funcionamento do jogo como um todo.

Na medida em que a ferramenta foi sendo desenvolvida, foram feitos testes funcionais das novas funcionalidades implementadas. Os testes desta etapa foram realizados por um grupo de estudantes do curso de Engenharia de Computação da UEFS que possuíam conhecimento em programação, sendo possível, desta forma, obter um retorno específico sobre possíveis erros, além de sugestões para a implementação de novos requisitos.

Esta seção tem por finalidade apresentar, de forma breve, a evolução da ferramenta BEM, já iniciada, de certa maneira, na seção anterior de metodologia, e, em seguida, detalhar o BEM Expressões, foco principal deste artigo.

5.1 A evolução do bem

O *Blinds, Education and Mathematics*, ou mais popularmente chamado de BEM, foi idealizado inicialmente por Dantas (2013), através do seu trabalho de conclusão do curso de Engenharia de Computação na Universidade Estadual de Feira de Santana. O mesmo propunha uma ferramenta que funcionasse como um objeto de aprendizagem, capaz de auxiliar pessoas no processo de aprendizagem da matemática, promovendo ainda a inclusão sócio digital de indivíduos com deficiência visual.

Na conclusão de sua primeira versão, o objeto de aprendizagem foi testado pelo público alvo (pessoas com DV), identificando-se a necessidade de algumas melhorias, tais como: inclusão da operação e do jogo da divisão, melhor qualidade na reprodução da voz e inserção de níveis de dificuldade.

Identificadas estas melhorias, foi iniciado o processo de construção da segunda versão do BEM (MACÊDO et. al., 2015), integrando-se à ferramenta o jogo de divisão, melhorias relacionadas à síntese de voz e os níveis de dificuldade, relacionados ao tamanho do tabuleiro.

Percebendo o potencial da ferramenta na educação matemática, uma nova versão foi lançada, expandindo as operações matemáticas para além de apenas dois operandos e uma só operação, para o tratamento de expressões matemáticas (MACÊDO et. al., 2016). Essa nova versão continha um novo módulo voltado para decidir, a partir de um valor/resultado dado, que expressão correspondia àquele valor.

SANTOS e outros (2016) realizaram uma nova avaliação do estado da ferramenta naquela versão, e, com ela, novas sugestões foram dadas pelos usuários DV para melhoria do BEM, em especial, aquelas que dão ao jogo mais dinâmica, ludicidade e atratividade, tais como os aspectos de *gamification*, de jogar contra um adversário e não sozinho, e a possibilidade de habilitar e desabilitar a contagem de tempo regressiva como mais um nível de dificuldade. Em Macêdo e outros (2017), a ferramenta ganhou uma nova funcionalidade voltada para melhorar a inclusão e a acessibilidade do público com deficiência visual, que foi o já mencionado uso da tecnologia de reconhecimento de voz. Comandos de voz poderiam ser pronunciados, através de um microfone, para a ferramenta, permitindo mais uma forma de troca de informações entre o jogador e o jogo (máquina).

Através dos testes aplicados com a ferramenta na versão *desktop*, não apenas com videntes, como também com pessoas com deficiência visual, percebia-se que a

sua instalação, mesmo com a utilização de tutoriais, poderia ser dificultosa para leigos. Surge, então, a proposta da nova versão, chamada de BEM Expressões Aritméticas, executada em navegadores *WEB* (e.g. *Google Chrome*, *Mozilla Firefox*, entre outros). Essa versão dispensa do usuário a necessidade de instalação do *software* e dos recursos associados, o que na versão anterior era uma limitação.

5.2 Bem: expressões aritméticas

O jogo BEM, em sua versão contendo expressões aritméticas, é um *software* que se enquadra na categoria de objetos de aprendizagem e tem como foco crianças com deficiência visual e videntes, dentro da faixa etária a partir dos 11 anos.

Este objeto de aprendizagem permite que pessoas com deficiência visual o utilizem a partir dos sentidos que os mesmos dispõem, como o tato e a audição, através de teclas de atalho e da síntese de voz, recurso que converte texto em áudio. Vale ressaltar que todo o ambiente de jogo pode ser acessado também através do mouse, e, diferente da versão do BEM que não contém expressões aritméticas, não há reconhecimento de voz.

Em sua versão inicial (Figura 1), o BEM Expressões tinha como plataforma de utilização o *desktop*, e se fazia necessária a instalação de componentes para que o jogo funcionasse corretamente.

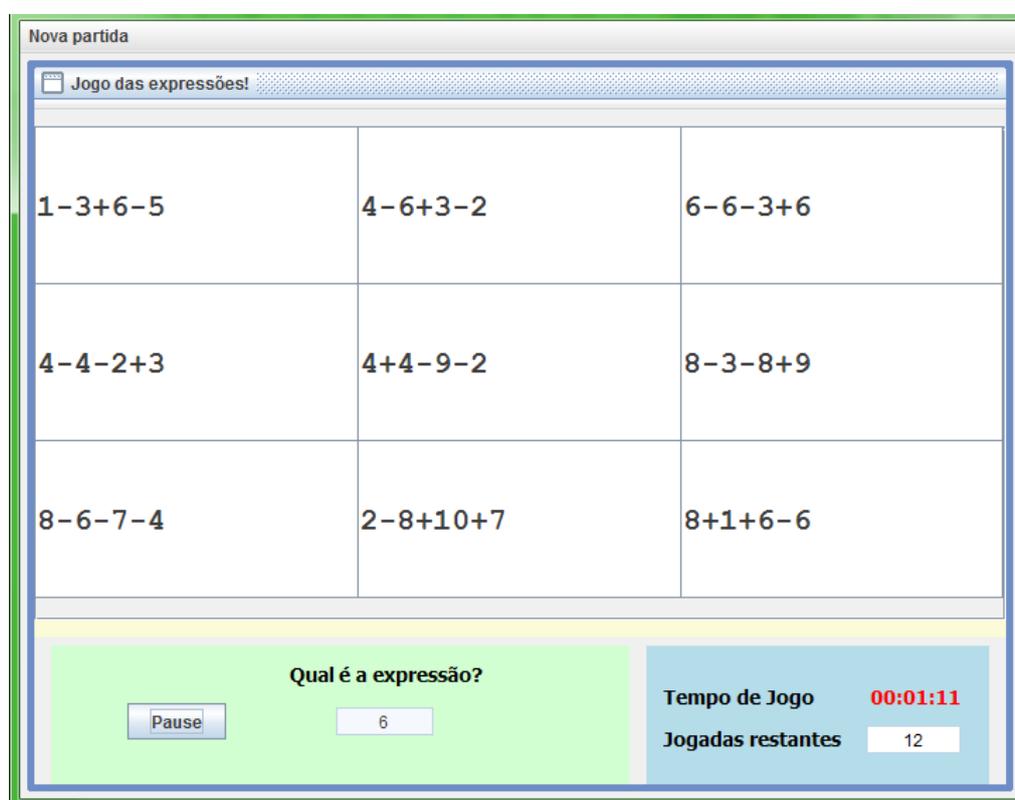


Figura 1 - BEM expressões versão desktop

Fonte: Próprio autor (2018)

O jogo BEM Expressões conta com alguns elementos que têm como objetivo tornar a experiência de seu uso mais atrativa, tais como níveis de dificuldade e a

contagem do tempo gasto pelo usuário para finalizar o jogo.

Com o passar do tempo, foi notado que o processo de instalação, que não era trivial, poderia afastar os usuários e desincentivar o uso do jogo. Tendo isso em mente, uma nova versão do jogo foi implementada para funcionar em navegadores *web*, de modo a remover qualquer processo de instalação ou possível dificuldade para o acesso ao jogo.

A dinâmica do jogo funciona da seguinte forma: é dado ao usuário um tabuleiro repleto de expressões aritméticas aleatórias com operandos na faixa de 1 a 10 (Figura 2 - A). O objetivo do usuário é encontrar a expressão correta para o resultado que é dado (Figura 2 - D). Além disso, há um cronômetro (Figura 2 - F) para estimular o jogador a resolver as expressões no menor tempo possível, juntamente com um contador de jogadas, responsável por evitar que o usuário tente acertar a expressão correta com a estratégia de “tentativa e erro”. Dessa forma, a cada jogada feita, uma unidade é consumida, sendo um total de 12 jogadas (Figura 2 - E). Quando os três acertos consecutivos são computados, o usuário ganha uma nova jogada, como forma de estímulo para escolher somente as expressões corretas.

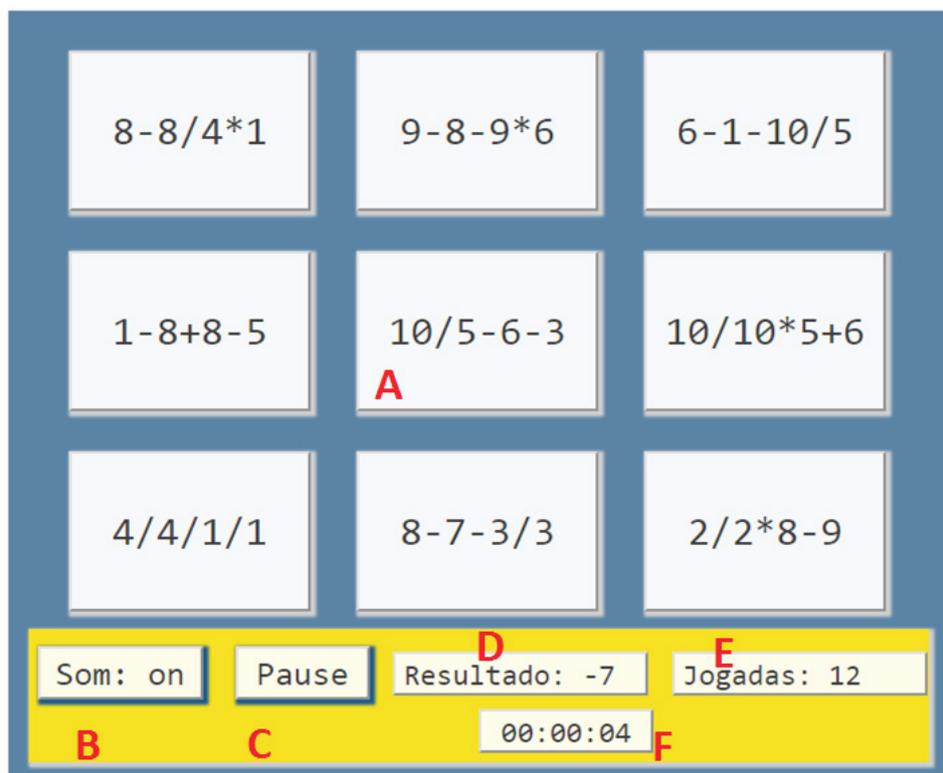


Figura 2 - BEM Expressões versão web

Fonte: Próprio autor (2018)

O usuário ainda pode ativar ou desativar o recurso de síntese de voz (Figura 2 - B). Nesse caso, quando o recurso é desligado, efeitos visuais são ativados automaticamente para casos de acertos e erros (Figura 3 - B). Existe também a opção de pausar o jogo (Figura 2 - C), e, quando pausado, todas as expressões são

escondidas para evitar que o jogador as resolva enquanto o tempo está pausado (Figura 3 - A).

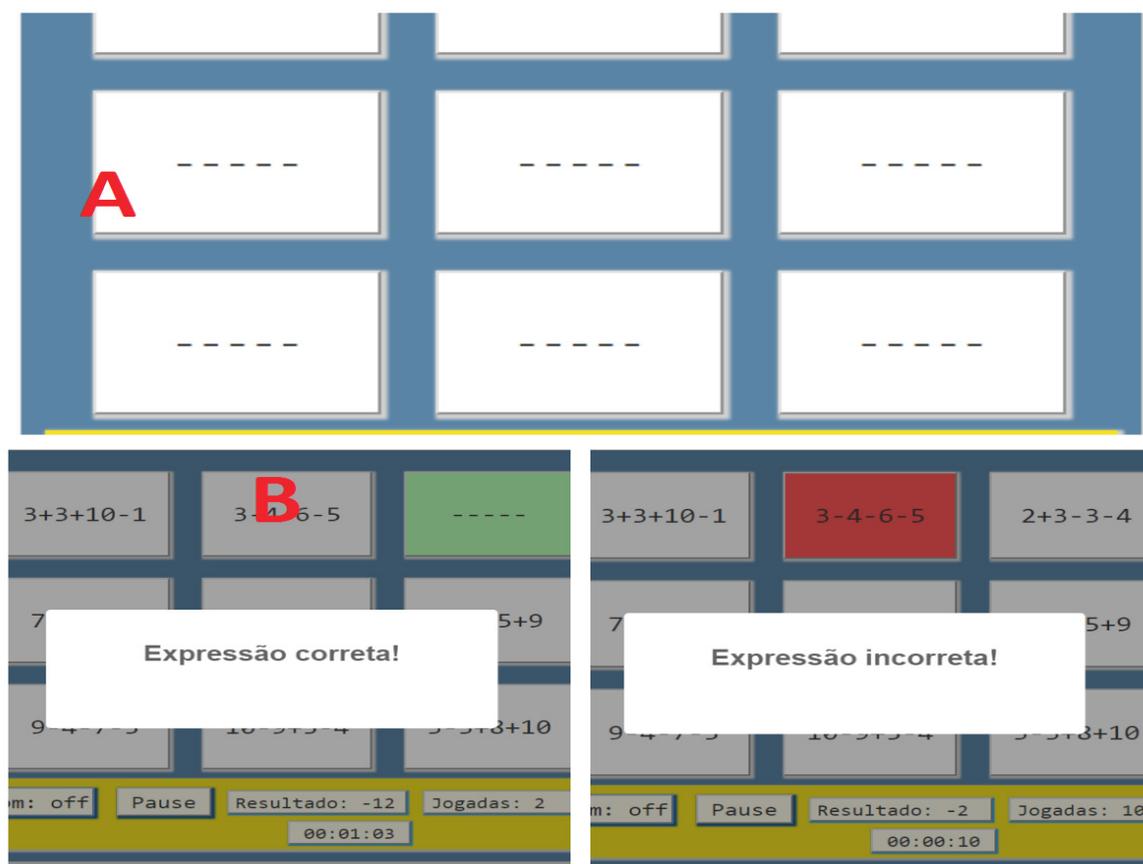


Figura 3 - Efeitos visuais do BEM Expressões

Fonte: Próprio autor (2018)

Os níveis de dificuldade podem ser escolhidos na tela inicial do jogo (Figura 4) ou a partir de teclas de atalho. Existem três níveis de dificuldade: fácil, intermediário e difícil. Tais níveis trabalham a quantidade de operadores que pode surgir nas expressões. No nível fácil, somente operadores de soma e subtração podem surgir; no nível intermediário, operadores de soma, subtração e multiplicação podem aparecer nas expressões; no nível difícil, todos os 4 operadores básicos podem ser utilizados, a qualquer momento, nas expressões. O intervalo de operandos se mantém de 1 a 9 em todos os níveis, bem como a quantidade de células do tabuleiro, sempre em número de 9. Há também um sistema de *ranking* dividido por nível de jogo, o que significa que o melhor tempo de cada nível é exibido na tela inicial.

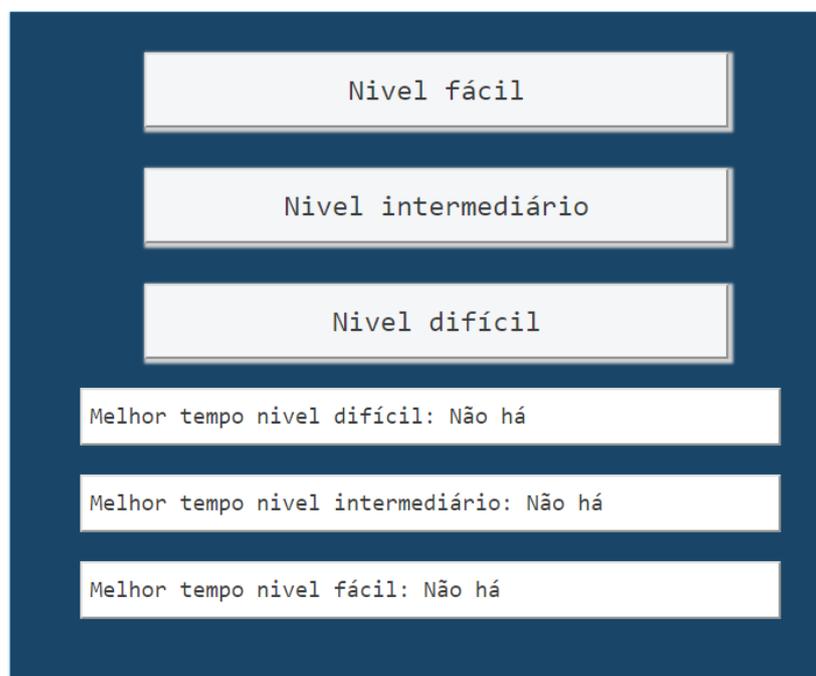


Figura 4 - Tela inicial do jogo BEM Expressões

Fonte: Próprio autor (2018)

Todas as opções disponíveis no jogo podem ser acessadas a partir de teclas de atalho presentes no teclado convencional de qualquer computador, conforme o Quadro 1, garantindo à pessoa com DV navegabilidade pelo sistema e, conseqüente, acessibilidade.

Atalho	Ação
Tecla 'a'	Inicia o nível fácil de jogo
Tecla 's'	Inicia o nível intermediário de jogo
Tecla 'd'	Inicia o nível difícil de jogo
Direcional 'cima'	Pausa o jogo
Direcional 'esquerda'	Sintetiza o resultado da expressão atual esperada
Direcional 'baixo'	Sintetiza a quantidade de jogadas restantes
Direcional 'direita'	Sintetiza o tempo atual de jogo
Tecla 'r'	Sintetiza o ranking do jogo
Tecla 'ctrl'	Ativa/Desativa a síntese de voz
Tecla 'w'	Interrompe qualquer síntese de voz que esteja em andamento
Tecla 'esc'	Volta à tela inicial
Tecla 'h'	Sintetiza todas as teclas de atalho de menu e suas respectivas ações
Tecla 'j'	Sintetiza uma breve explicação sobre o funcionamento dos atalhos para o tabuleiro de expressões
Tecla 'q'	Sintetiza um breve tutorial sobre o jogo

Quadro 1 - Relação de atalhos do BEM Expressões

Fonte: Próprio autor (2018)

Para acessar as células do tabuleiro, o usuário utiliza o teclado numérico do computador, de maneira análoga ao tabuleiro (Figura 5). Dessa forma, o usuário com deficiência visual pode se localizar no tabuleiro e ir diretamente à célula que deseja. Por exemplo, caso o usuário deseje escolher a célula central do tabuleiro, ele deve pressionar a tecla de número 5 em seu teclado numérico. Após pressionar a tecla correspondente à expressão desejada, há, primeiramente, a síntese da expressão para o usuário, para que ele possa ouvi-la e relacioná-la à célula do tabuleiro (alguns deles conseguem, inclusive, memorizar este posicionamento). Caso o mesmo deseje, após a escuta, selecioná-la, basta pressionar novamente a mesma tecla para confirmação.

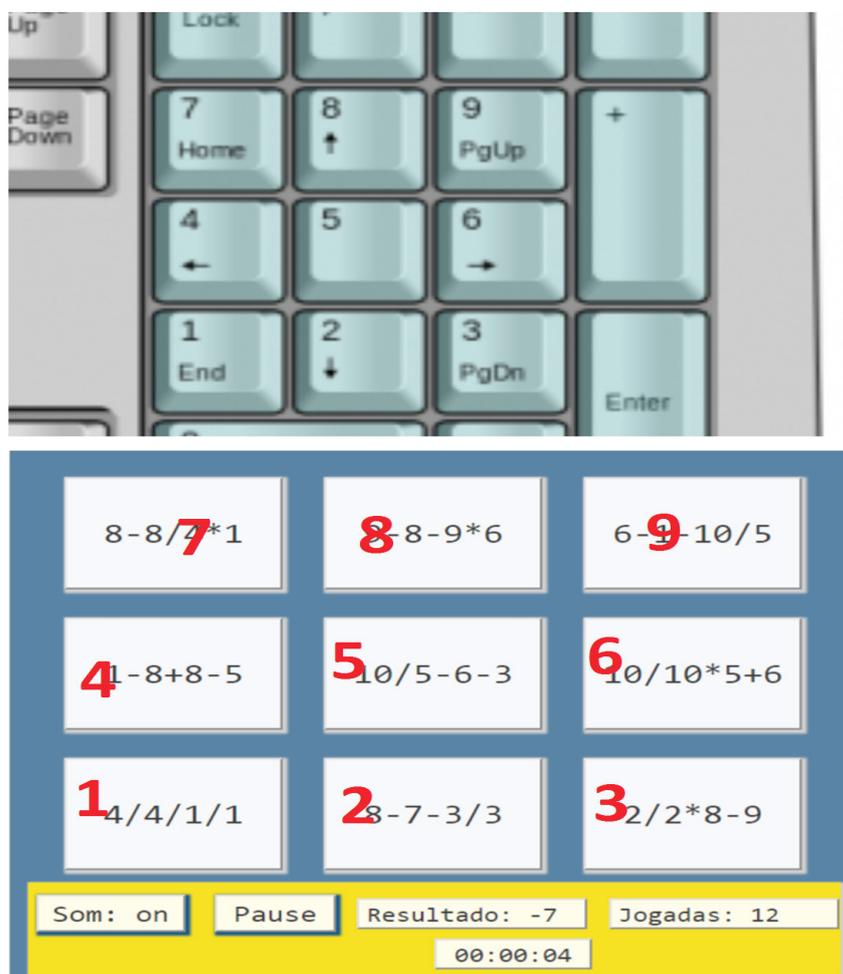


Figura 5 - Mapeamento do tabuleiro do jogo

Fonte: Próprio autor (2018)

6 | CONSIDERAÇÕES FINAIS

O jogo BEM - Expressões apresenta potencialidades para o auxílio no processo de ensino aprendizagem da matemática, tanto na perspectiva do educando, quanto na perspectiva do educador, que pode planejar estratégias metodológicas com o uso

da ferramenta dentro e fora do espaço escolar. A partir desta ferramenta, é possível, além de exercitar as quatro operações básicas da matemática de forma conjunta (expressões aritméticas), também concretizar a inclusão de pessoas com deficiência visual, tanto na perspectiva social quanto na digital.

Considerando que os avanços das TIC permeiam diversas áreas de conhecimento e vários setores da sociedade, e que a educação, como espaço de (in)formação, não pode desconsiderar este cenário atual, permitir que as pessoas com DV tenham a oportunidade de utilizar um programa de computador que visa o avanço cognitivo e pessoal, é ampliar as habilidades desses cidadãos, tanto no manuseio das tecnologias digitais, quanto nos conceitos lógico matemáticos. A inclusão digital os aproxima do contexto das novas tecnologias e, inevitavelmente, promove também a inclusão social, uma vez que permite uma interação entre eles e seus pares, ou ainda, com os docentes e/ou com outras pessoas que se envolvam no jogo e troquem experiências. Além disso, há também a valorização do aprender e do interagir com o mundo e com as TIC a partir dos sentidos remanescentes que possuem, como a audição e o tato. Enquanto tecnologia assistiva, há a preocupação, no jogo, com a acessibilidade, e em garantir que tanto os DV quanto os videntes possam utilizá-lo de forma confortável.

Atualmente, o BEM Expressões não oferece suporte ao reconhecimento de voz, contudo, isso não prejudica sua utilização pelo público com deficiência visual, que tem o teclado como modalidade de entrada e a voz, como recurso de saída. Todas as ações do jogo podem, portanto, ser controladas a partir de teclas de atalho, presentes em qualquer teclado comum de um computador. Dessa forma, almeja-se, como trabalhos futuros, a implementação do suporte do reconhecimento de voz no BEM Expressões, a distribuição do *software* em instituições de ensino e a realização de testes com o público alvo, a fim de avaliar, entre outros aspectos, a qualidade do reconhecimento de voz e a efetividade da aplicação do *software* BEM como recurso pedagógico na sala de aula. Além disso, pretende-se ampliar a aplicação de testes, envolvendo docentes e discentes, a fim da real avaliação da potencialidade educacional da ferramenta.

REFERÊNCIAS

ALVES, C. P. Fala Brasil: Reconhecimento de Voz para o Português Brasileiro, 2015. Disponível em: <<http://www.laps.ufpa.br/falabrasil/>>. Acesso em: 10 de abril de 2018.

BERSCH, R. Introdução à tecnologia assistiva. Porto Alegre: CEDI, p. 21, 2008.

CLOUDGARDEN. Disponível em: <<http://www.cloudgarden.org/JSAPI/>>. Acesso em: 19 de abril de 2018.

CMUSPHINX. Sphinx 4, 2015. Disponível em: <<http://cmusphinx.sourceforge.net/wiki/sphinx4:webhome>>. Acesso em: 18 de abril de 2018.

DANTAS, A. L. P. Bem: um objeto de aprendizagem para apoiar crianças deficientes visuais e videntes no processo educacional das operações básicas de matemática. 2013. 105 f. Trabalho de Conclusão

de Curso (Engenharia de Computação) - Universidade Estadual de Feira de Santana, Feira de Santana, 2013.

GIROTO, C. R. M.; POKER, R. B.; OMOTE, S. As tecnologias nas práticas pedagógicas inclusivas. Oficina Universitária, Universidade estadual paulista, faculdade de filosofia e ciências, São Paulo-SP, 2012.

HEHIR, T.; GRINDAL, T.; FREEMAN, B.; LAMOREAU, R.; BORQUAYE, Y.; BURKE, S. Os Benefícios da Educação Inclusiva para Estudantes com e sem Deficiência. Instituto Alana, Abt Associates, Pinheiros, São Paulo – SP, 2016. Disponível em: http://alana.org.br/wpcontent/uploads/2016/11/Os_Beneficios_da_Ed_Inclusiva_final.pdf, Acesso em: 20 de abril de 2018.

Techtudo, IBM Embedded ViaVoice, em: <http://www.techtudo.com.br/tudo-sobre/ibm-embedded-viavoice.html> >. Acesso em: 17 de abril de 2018.

MACEDO, R. S.; FERNANDES, A. L. B.; SENA, C. P. P. ; SANTANA, K. C.; SANTOS, A. J. O. S.. Blinds, Education and Mathematics: jogo eletrônico de tabuleiro como estratégia educativa para o ensino aprendizagem das quatro operações básicas da matemática. In: ERBASE, 2016, Maceió. WEIBASE, 2016.

MACEDO, R. S.; SENA, C. P. P.; SANTOS, A. J. O. S.; PINTO, G. R. P. R. ; FERNANDES, A. L. B. . A utilização de jogos na educação matemática para crianças com deficiência visual e videntes. In: ERBASE, 2015, Salvador. WTICGBASE, 2015.

MACÊDO, R.; PEREIRA, C. P. P.; FERNANDES, A.; SANTANA, K.; SANTOS, A. J. O. S. Blinds, Education And Mathematics: Objeto De Aprendizagem Sobre As Operações Básicas Da Matemática Com O Uso Dos Recursos De Síntese E Reconhecimento De Voz. In: XXVIII Simpósio Brasileiro de Informática na Educação SBIE (Brazilian Symposium on Computers in Education), 2017, Recife, 2017. p. 445.

OLIVEIRA, A. M., LUDWIG, L., FINCO, M.D. Proposta Pedagógica do Uso das TICs como Recurso Interdisciplinar. Anais do XXII SBIE - XVII WIE. Aracaju, 21 a 25 de novembro de 2011. ISSN: 2176-4301

Pinto, A. V. Sete lições sobre educação de adultos. 8ª edição. São Paulo, Cortez, 1993. ISBN 85-249-0202-7, 2011.

PASSERINO, L. M.; MONTARDO, S. P. Inclusão social via acessibilidade digital: proposta de inclusão digital para pessoas com necessidades especiais. Revista da Associação Nacional dos Programas de Pós-Graduação em Comunicação, v. 8, p. 1-18, 2007.

PELOSI, M. B. et al. Caracterização dos professores itinerantes, suas ações na área de tecnologia assistiva e seu papel como agente de inclusão escolar. Revista Brasileira de Educação Especial, v. 15, n. 1, p. 141-154, 2009.

PINHEIRO, N. A. M.; SILVEIRA, R. M. C. F.; BAZZO, W. A. ciência, Tecnologia e Sociedade: A Relevância do Enfoque CTS Para o Contexto do Ensino Médio. Revista Activa, Universidade Estadual Paulista Júlio de Mesquita Filho Brasil, v. 13, n. 1, p. 71-84, jan. 2007.

RESPONSIVEVOICE. Disponível em: <https://responsivevoice.org/api/>>. Acesso em: 22 de abril de 2018.

ROCHA, A. N. D. C.; DELIBERATO, D. Tecnologia assistiva para a criança com paralisia cerebral na escola: identificação das necessidades. Revista Brasileira de Educação Especial, p. 71-92, 2012.

SANTANA, K.; PEREIRA, C. P. P. ; FERNANDES, A. ; SANTOS, A. J. O. S.; MACEDO, R. Blinds, Basic Education: jogo digital inclusivo para auxiliar o processo de ensino-aprendizagem das pessoas com deficiência visual. In: XXVIII Simpósio Brasileiro de Informática na Educação SBIE (Brazilian Symposium on Computers in Education), 2017, Recife, 2017. p. 877.

SANTOS, A. J. O. S.; FERNANDES, A. L. B.; SENA, C. P. P.; SANTANA, K. C.; MACEDO, R.

S.. Análise do uso do Objeto de Aprendizagem BEM no processo educacional de pessoas com deficiência visual e videntes. In: ERBASE, 2016, Maceió. WTICGBASE, 2016.

SILVA, L. M., Educação inclusiva e formação de professores, MEC – SETEC. Instituto federal de educação, ciência e tecnologia do estado de Mato Grosso, Cuiabá - MT, 2009.

SOUZA, R. S. E. Indisciplinaridade na educação de infância: a roda olímpica do movimento, expressão, corpo e ludicidade – Dissertação (mestrado) - Universidade Federal de Mato Grosso do Sul. Centro de Ciências Humanas e Sociais. Campo Grande, MS, 2009.

JINDIE: UMA LINHA DE PRODUTO DE SOFTWARE PARA JOGOS EDUCATIVOS COM FOCO NO CONSTRUCIONISMO

Carlos Alberto Correia Lessa Filho

Centro Universitário CESMAC

Maceió – Alagoas

Arturo Hernandez Dominguez

Instituto de Computação/Universidade

Federal de Alagoas

Maceió – Alagoas

RESUMO Os jogos no processo de ensino têm se demonstrado uma interessante ferramenta para contribuir com a motivação e aprendizado dos estudantes. Este trabalho tem como objeto facilitar o desenvolvimento desses jogos educativos. Para isto, foi desenvolvida uma Linha de Produto de Software com base em jogos construcionistas. Essa Linha de Produto foi utilizada na produção de quatro jogos. Baseando-se nos resultados, pode-se observar as vantagens no uso de uma Linha de Produto no desenvolvimento de jogos educativos como redução no tempo de desenvolvimento, simplificação da complexidade e direcionamento do processo de criação dos jogos.

PALAVRAS-CHAVE jogos educacionais, Linha de Produto de Software, motivação

ABSTRACT The games in the teaching process have proved an interesting tool to contribute with the motivation and student learning. This paper aims to facilitate the development of these

educational games. To do this, we developed a Software Product Line based on constructionist games. This Product Line was used to develop four games for its validation. Based on the results, it's possible to realize the advantages in using a Product Line in the development of educational games, such as, reduction in development time, simplification of code's complexity and directing of the creation of games.

KEYWORDS educational games, Software Product Line, motivation

1 | INTRODUÇÃO

Nos últimos anos, muitas pesquisas na educação têm sido realizadas com o foco em elevar a motivação dos estudantes para buscar novos conhecimentos. Várias dessas pesquisas têm apontado a diversão como um fator importante para alavancar a motivação dos alunos aos estudos (YOON; KIM, 2015). Atualmente, os jogos vêm ganhando importância na área de ensino, de forma que algumas instituições estão começando a adotá-los em sala de aula, com a intenção de despertar a motivação aos estudos que os estudantes não possuíam, como também possibilitar novas experiências, que antes eram difíceis sem o computador (REZENDE, et al., 2013).

A abordagem tradicional de ensino entre professores e estudantes, onde apenas o professor transmite o conteúdo, enquanto os estudantes a decoram, também se demonstra defasada e desmotivadora, no qual cerca de 40% dos estudantes entre 15 e 17 anos que deixam os estudos, apontam a escola como algo desinteressante (SANTOS et al, 2015). Papert e Harel (1991) desde antigamente defendiam a motivação no processo de ensino através do construcionismo, no qual o estudante para aprender um conteúdo por completo e ser motivado pelo conteúdo estudado, deve participar da construção de um material concreto. Papert ainda defende a importância dos jogos nesse processo, onde uma criança apresenta menos resistência ao conteúdo sendo apresentado em um jogo comparado ao conteúdo apresentado de forma tradicional em sala de aula.

Os jogos educativos podem contribuir bastante com o processo de resgate do interesse dos estudantes, de forma que o ideal seria que professores pudessem desenvolver seus próprios jogos (TAROUCO et al, 2004). Todavia, o processo de desenvolvimento de um jogo do zero é um processo demorado, caro, que pode envolver muitas pessoas. Para tentar contornar essa situação, o desenvolvedor pode optar pelo uso de uma Linha de Produto de Software (LPS). Uma LPS segundo Käköla e Leitner (2014) é uma metodologia validada industrialmente para o desenvolvimento de *softwares* visando menor custo, maior velocidade, qualidade e satisfação do usuário final.

Visando a importância desses jogos e a escassez de recursos para o desenvolvimento de jogos no contexto do construcionismo, este artigo apresentar uma proposta de uma LPS para o desenvolvimento de jogos construcionistas seguindo o modelo proposto por Pohl, Böckle e Linden (2005). A avaliação da LPS construída foi fundamentada no desenvolvimento de quatro jogos, sendo estes baseados em três jogos construcionistas.

2 | TRABALHOS CORRELATOS

Nessa seção iremos apresentar duas LPS, que foram selecionadas por também serem voltadas para o desenvolvimento de jogos ou aplicativos educativos.

A LPS Arcade Game Maker (MORIN et al, 2008) foi criada pelo *Software Engineering Institute* (SEI) com o intuito de criar um exemplo de Linha de Produto para dar suporte a aprendizagem e experiência em Linha de Produtos de Software. A linha de produto conta com a presença de três jogos do gênero plataforma. Ao contrário da maioria dos jogos, os produtos do Arcade Game Maker não possuem foco na interface gráfica, pois o objetivo desta LPS é ser um exemplo abrangente. Seu material sofreu evoluções de pelo menos dois anos de produção e está sendo usado como exemplo ilustrativo em cursos.

A segunda LPS apresentada é a Linha de Produtos de Software para Módulos de Aprendizagem Interativa (iMA) (DALMON; BRANDÃO, 2013). iMA são módulos educativos, como jogos, que são integrados a ambientes educacionais como o Moodle. A LPS de Dalmon e Brandão foi desenvolvida com a preocupação em reduzir as falhas e o tempo gasto para resolver tais problemas no desenvolvimento de iMA, que geram impactos diretos nos benefícios didáticos que esses módulos pretendiam fornecer. O desenvolvimento da LPS foi baseado nos iMA já existentes.

A análise dessas e outras LPS baseadas em aplicativos e jogos educativos auxiliaram no processo de definição dos recursos e métodos de implementação da LPS proposta. A LPS deste trabalho diferencia-se desses e outros trabalhos correlatos ao especializa-se nos recursos usados por jogos do domínio construcionista, que não dispõem de uma ferramenta especializada conhecida.

3 | CONSTRUCIONISMO

O construcionismo de Seymour Papert foi baseado no construtivismo de Jean Piaget, que estudou as fases do conhecimento com uma visão biológica. Para Piaget, o aprendizado ocorre de diferentes formas ao longo da vida do indivíduo, sendo dividida em quatro etapas: Sensório-motor (até 2 anos); Pré-operatório (até 7 anos); Operatório-concreto (até 12 anos); Operações formais (após os 12 anos) (MARLOWE; CANESTARI, 2006).

Para Papert, o construcionismo não apenas compreende as fases do conhecimento proposto por Jean Piaget, como também considera que o sujeito deve construir um material concreto como, por exemplo, uma apresentação, um quadro ou um *software*. Assim, novos conhecimentos serão adquiridos através da construção. Para Papert o desenvolvimento de um material concreto é tão importante quanto o pensamento abstrato (KAFAI, 2006). Nesse processo de construção será feita uma conexão entre o concreto e o abstrato por meio de reflexões, possibilitando o sujeito pôr em prática suas ideias, teorias e hipóteses, que antes eram apenas trabalhadas de forma abstrata. Um conceito antigo no construcionismo está nos erros cometidos no processo de construção, no qual não terá mais um papel de punição, pois através dos erros será preciso compreender os equívocos cometidos, reformular seu processo de reflexão e criação para obter o resultado desejado (LEBOW, 1993). Apenas após todos esses procedimentos de hipóteses, prática e reflexão é que poderá chegar ao conhecimento real sobre um determinado assunto.

O papel do computador no construcionismo tem uma grande importância, por facilitar esse trabalho de construção, uma vez que em determinadas situações se tornaria inviável construir algo na vida real, além de possibilitar situações lúdicas que tornem a aprendizagem prazerosa segundo Papert (PAPERT; FREIRE, 2000).

4 | LINHA DE PRODUTO DE SOFTWARE

O conceito de Linha de Produto de Software teve início com a ideia de Linha de Produção de Henry Ford no setor automobilístico. A ideia de Linha de Produção consistia em criar uma sequência de etapas repetidas e produzida pelas mesmas pessoas as tornando especialistas nessas atividades (TURI et al, 2015). Todavia, o produto final seguia sempre o mesmo modelo nessa produção em massa. Com o tempo surgiu à necessidade de atender diferentes requisitos para cada cliente, como cores de automóvel e tamanho diferentes. Essa necessidade originou a customização em massa, onde inicialmente é criada uma plataforma em comum, que possui todas as características comuns daquele tipo de produto e apenas posteriormente eram realizadas as modificações exigidas por cada cliente para gerar o produto final finalizado. Dessa ideia surgiu a Linha de Produto de *Software*, onde se pretende criar uma plataforma em comum com todos os *softwares* de um determinado domínio, com pontos flexíveis. A plataforma comum possui as características chamadas comunalidades, que estão presentes em quase todos os *softwares* do domínio, e deve permitir a flexibilidade, chamada de variabilidade, que são os pontos onde o *software* pode ser customizado para atender a necessidade do cliente (POHL et al, 2015).

O processo de desenvolvimento de um produto em uma LPS é dividido em duas partes (BLANES; INSFRAN, 2012): Engenharia de Domínio – Estabelece a plataforma de reutilização e define as comunalidades e variabilidades; Engenharia de Aplicação – Utiliza-se a arquitetura e processo do domínio para gerar uma aplicação final usando as comunalidades e selecionando as variabilidades.

A reutilização no contexto de uma LPS não está apenas na reutilização de código e sim na reutilização do processo e da arquitetura. A reutilização do código ocorre quando há o reaproveitamento de um código já implementado anteriormente; reutilização do processo ocorre quando atividades, pessoas e papéis são reaproveitados; e a reutilização de arquitetura ocorre quando uma estrutura complexa é utilizada para resolver um determinado problema (DALMON; BRANDÃO, 2013).

Para possibilitar a implementação e seleção da variabilidade na plataforma comum, pode se adotar diversas técnicas de programação como agregação e delegação, parametrização, herança, *overloading*, compilação condicional, programação orientada a aspectos ou padrões de projetos (GACEK; ANASTASOPOULES, 2001). O desenvolvedor ainda pode contar com ferramentas para a produção de LPS como o *Colored Integrated Development Environment* (CIDE) (FEIGENSPAN et al, 2010).

5 | METODOLOGIA

Inicialmente foi trabalhado o desenvolvimento de um jogo construcionista para

então nesta nova etapa, iniciar o desenvolvimento de uma LPS voltada para esses jogos.

No contexto da LPS proposta nesse artigo, foi tido como base o modelo apresentado por Pohl, Böckle e Linden (2005). Esse modelo é um dos mais completos e utilizados em vários cursos e artigos. Entretanto, assim como é sugerido, esse modelo foi adaptado para atender as reais necessidades desse trabalho.

Para o trabalho voltado ao desenvolvimento da Engenharia de Domínio, foram selecionados 10 jogos construcionistas analisando tanto as funcionalidades do jogo, quanto os artefatos construídos nesses jogos. A seleção do número de jogos buscou atender diferentes tipos de jogos construcionistas sem repetições e que não comprometessem com o tempo disponível para a produção da LPS. Na implementação do código da plataforma comum, foram gerados dois *frameworks*, sendo um desenvolvido em PHP sem o uso de ferramentas de apoio e um segundo *framework* desenvolvido em Java utilizando a ferramenta de suporte à construção de LPS, CIDE. O desenvolvimento dos dois *frameworks* serviu para verificar se o modelo proposto está apto a realizar diferentes *frameworks*, avaliar diferentes técnicas de implementação da variabilidade e o processo realizado com e sem o uso de uma ferramenta de suporte a produção de LPS. Para a avaliação da LPS desenvolvida na Engenharia de Domínio, foram desenvolvidos quatro jogos na Engenharia de Aplicação, sendo 3 deles desenvolvidos na versão em PHP e um quarto na versão em Java. A avaliação do processo de desenvolvimento dos jogos foi baseada na verificação da viabilidade de construção de jogos, nas métricas de número de linhas de código, tempo de desenvolvimento, complexidade ciclomática e opinião do desenvolvedor dos jogos. Na avaliação também são realizadas comparações de um jogo desenvolvido com a LPS e sem a LPS desenvolvida, nomeada de JIndie, como também o desenvolvimento através da técnica de compilação condicional com a ferramenta CIDE e outra versão sem apoio de uma ferramenta de construção de LPS.

O desenvolvimento da LPS pode ser visto na seção 6 deste artigo e sua aplicação no desenvolvimento de jogos é apresentada na seção 7, seguida dos resultados na seção 8.

6 | UMA LINHA DE PRODUTO DE SOFTWARE PROPOSTA NO CONTEXTO DE JOGOS CONSTRUCIONISTAS

Nesta seção será apresentada a LPS proposta para o desenvolvimento de jogos construcionistas. A LPS foi intitulada de JIndie, originada do termo Jogos Indie, normalmente utilizados para jogos desenvolvidos por pequenas empresas ou pessoas sem apoio financeiro. A abordagem nesta seção será da Engenharia do Domínio, passando pelas etapas de identificação e resolução dos problemas.

Um problema numa LPS é identificado como sendo os requisitos que os

softwares necessitam, também conhecidos como *features* ou as comunalidades e variabilidades da LPS. Devido ao fato de não possuir *stakeholders* para levantar os requisitos, a ação realizada na etapa de levantamento dos requisitos foi semelhante a LPS iMA (DALMON; BRANDÃO, 2013), utilizando como base *softwares* e jogos já existentes. Nesta primeira etapa foram selecionados 10 jogos construcionistas e listados quais os requisitos presentes em cada um desses jogos tanto no contexto de um jogo construcionista de forma geral, como mais especificamente no artefato construído por esses jogos. Após a listagem dos requisitos, foram criadas duas tabelas de matriz de requisitos, uma para os requisitos do jogo e outra para os requisitos do artefato. A matriz de requisitos tem como finalidade filtrar e identificar quais desses recursos são requisitos exclusivos de um jogo, comunalidades ou variabilidades. Tendo filtrados e classificados os requisitos, estes foram organizados através do modelo *Feature Model*, que permite através de uma representação visual demonstrar as comunalidades, requisitos mandatórios, e variabilidades em forma de requisitos opcionais ou alternativos (FEIGENSPAN et al, 2010):

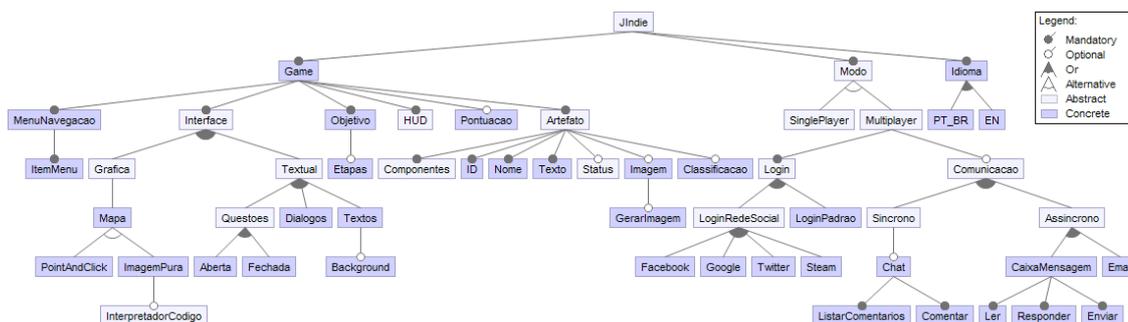


Figura 1 - *Feature Model* da LPS Jindie

Fonte: Dados da pesquisa

Entre os principais requisitos encontrados nos jogos construcionistas e representados na Figura 1, podemos citar a opção de criar imagens dinâmicas e interpretação de códigos inseridos pelo jogador como acontece no jogo Logo e RoboCode que serão apresentados na próxima seção, além de outros requisitos como comunicação entre jogadores, estilo *Multiplayer* ou *Single Player*, sistema de pontuação, escolha de idiomas e objetivos. Já no artefato o principal requisito identificado foi à necessidade de ter componentes que possam ser agregados e utilizados de forma fácil, para montar e finalizar o artefato sendo construído.

Para realizar essa integração foi planejado utilizar o Diagrama de Componentes, por facilitar a reusabilidade do artefato e seus componentes. Na Figura 2 podemos ver através do Diagrama de Componentes como quatro jogos distintos podem construir diferentes artefatos através do uso de interfaces. No primeiro bloco a direita (azul), podemos ver um jogo onde o artefato é um projeto de *software* ou uma empresa e o componente é um funcionário que está sendo contratado pela empresa. No segundo bloco (vermelho), o artefato sendo construído se trata de um bolo e o componente pode ser um ingrediente deste bolo, como açúcar. No terceiro bloco a direita (verde),

temos um jogo de construção de cidade, onde o componente adicionado é um prédio da cidade como hospital. Por fim no último bloco (laranja) temos um jogo de construção de imagens através de comandos, como acontece no jogo Logo. O componente Tile que aparece no diagrama representa blocos do desenho ou imagem sendo agregada junto com a adição do componente do artefato como o desenho de um prédio no mapa da cidade.

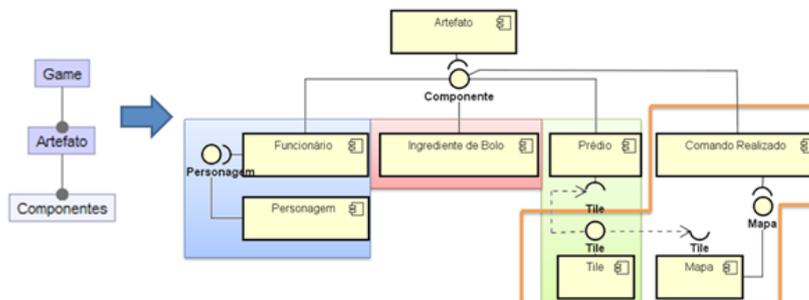


Figura 2 - Diagrama de Componente do Artefato e seus componentes
Fonte: Dados da pesquisa

Para a implementação da LPS foi adotada duas versões, na qual a primeira é composta por técnicas de implementação da variabilidade de agregação e delegação, parametrização, herança e técnicas de padrões de projetos com o *factory*. A segunda versão além de contar com essas técnicas citadas anteriormente, também foi desenvolvida com o suporte da ferramenta CIDE utilizando a técnica de implementação da variabilidade por compilação condicional. A escolha de duas implementações serviu para avaliar o impacto do uso de ferramentas de suporte a construção de LPS como o CIDE com a técnica de compilação condicional. A técnica de compilação condicional tem como finalidade definir quais trechos de códigos e arquivos pertencem a quais requisitos através do uso de *tags*, de forma que se um requisito não for necessário, todos os códigos e arquivos pertencentes a este, não estarão presentes na aplicação final.

7 | QUATRO JOGOS CONSTRUÍDOS A PARTIR DA LPS PROPOSTA

Para validar a viabilidade e qualidade do uso da LPS JIndie, foram desenvolvidos quatro jogos utilizando os dois *frameworks* da LPS. Os jogos desenvolvidos (Figura 3) foram baseados no Sim Investigador, RoboCode e Linguagem Logo que são jogos construcionistas que já foram utilizados em sala de aula.



Figura 3 - Jogos desenvolvidos através da LPS JIndie

Fonte: Dados da pesquisa

O primeiro jogo desenvolvido foi o Sim Investigador. Sim Investigador se trata de um jogo no qual o jogador assume o papel de detetive que deve solucionar casos envolvendo conteúdos de diversas disciplinas, como matemática, português, geografia ou história, através de um sistema de perguntas e respostas. Além de poder jogar um caso, o jogador também pode criar seus próprios casos assim como o professor também pode acompanhar os casos jogados e criados pelos seus estudantes (LESSA FILHO et al, 2015). No caso do desenvolvimento do Sim Investigador foi possível ter acesso ao código original assim como a opinião do desenvolvedor comparando o desenvolvimento com e sem a LPS JIndie.

O segundo jogo desenvolvido foi baseado na Linguagem Logo bastante utilizada em colégios no processo de ensino de formas geométricas. Neste jogo construcionista, o jogador assume o controle de uma tartaruga através de comandos como “parafrente” ou “paraesquerda”. A cada movimento realizado pela tartaruga na imagem, é deixando uma linha representando o caminho realizado, podendo assim criar figuras geométricas, num processo que busque a criatividade e reflexões de formas de figuras por parte do jogador (FEIN et al, 2013). O jogo desenvolvido com a LPS JIndie, além de contar com as funções de movimento, habilitar e desabilitar as linhas formadas ao mover a tartaruga, também apresenta a funcionalidade do jogador criar suas próprias funções como definir um método para criar um quadrado ou triângulo.

O terceiro e último jogo desenvolvido foi baseado no jogo RoboCode. O jogo RoboCode é um jogo voltado ao aprendizado de programação, permitindo aos jogadores através da linguagem Java ou C# criarem seus próprios robôs, para batalharem entre si (ROBOCODE, 2016). Este jogo foi desenvolvido duas vezes com a LPS JIndie, sendo uma delas assim com os jogos citados anteriormente sem a ferramenta de suporte a LPS, o CIDE, e a outra versão utilizando o CIDE e a técnica de compilação condicional. As duas versões desenvolvidas permitiram o jogo *on-line* utilizando o *login* pela rede social Facebook, possibilitando que os jogadores amigos pudessem batalhar entre si. O *login* por redes sociais é um dos recursos já disponibilizados pela LPS JIndie.

8 | RESULTADOS E DISCUSSÃO

Após realizar a produção dos jogos e observar que a LPS JIndie possibilita a construção de jogos construcionistas, foram analisados os jogos construídos através das métricas baseadas em trabalhos correlatos como número de linhas de códigos, opinião do desenvolvedor, necessidade de refatoração no domínio da aplicação, número de arquivos manipulados pelo desenvolvedor e a Complexidade Ciclomática (CC). A CC é uma métrica proposta por McCabe para avaliar a complexidade de *softwares* baseada nas linhas de códigos, de forma que através do fluxo do código é possível informar se este código apresenta baixo ou alto risco de apresentar falhas durante a sua manutenção (WATSON et al, 1996).

No desenvolvimento do primeiro jogo, Sim Investigador, foi possível realizar uma comparação com o desenvolvimento do código original. A implementação do jogo com a LPS não apresentou dificuldades que impedissem o desenvolvimento, pelo contrário, foi possível obter ganho de desempenho devido aos diversos recursos extras que a LPS oferece, possibilitando que o desenvolvedor se focasse no desenvolvimento do núcleo do seu jogo. O jogo original contém 6.937 linhas de código que foram implementadas pelo desenvolvedor, enquanto a versão utilizando a LPS apresenta 20.210 linhas, das quais apenas 1.849 foram implementadas pelo desenvolvedor, sendo o restante fornecido pela própria LPS, representando uma redução de 73,3% no esforço do desenvolvedor ao usar a LPS. O número maior de linhas no código está relacionado ao fato de que a arquitetura desenvolvida na LPS tende a ser mais genérica e completa para atender diversos tipos de jogos. Outros fatores que influenciam também no tamanho das linhas podem ser vistos em recursos de *logs*, filtros de segurança e *parses* de URL amigáveis, que não estavam presentes na versão original do jogo.

Ao avaliar a velocidade de carregamento de páginas entre a versão original e a versão com LPS, foi possível observar que a versão original é alguns milésimos mais rápidos que a LPS, todavia esses milésimos são refletidos no fato da LPS possuir mais filtros e controles de segurança a cada solicitação. Em relação à análise da CC desses códigos foi possível observar um ganho considerável na qualidade dos códigos, de forma que os códigos com a LPS se tornaram mais limpos e menos complexos. Quanto menor a CC, mais fácil é de trabalhar no código e quanto maior, mais complexo esse código é. Recomenda-se que este valor esteja abaixo dos 20 pontos. No código original no cadastro dos usuários é possível ver uma CC com 14 pontos, enquanto na versão com a LPS JIndie esse valor caiu para apenas 1 ponto. No geral o desenvolvimento do Sim Investigador com a LPS se tornou mais organizado e robusto devido a padrões como MVC. Seu processo foi de fácil adaptação, obtendo um ganho de 45% do tempo usado na implementação do jogo. Para essa avaliação, foram desconsiderados o tempo de definição dos requisitos e criação da interface do jogo, uma vez que já haviam sido desenvolvidos e definidos no jogo original.

Na produção do jogo Logo foi possível realizar a conclusão do jogo com o desenvolvimento de apenas 409 linhas por parte do desenvolvedor (10,01% de todo código usado no jogo, sendo o restante disponibilizado pela LPS), uma vez que os recursos mais complexos como gerar imagem e interpretar códigos inseridos pelo jogador, já são disponibilizados pela própria LPS. Todos seus códigos apresentaram CC abaixo de 5 e sua produção foi possível ser realizada em apenas 16h. Todavia, a LPS apresentou uma limitação relacionada a imagem gerada. A classe responsável por gerar as imagens é baseada em um desenho formado por blocos como uma matriz, o que limitava a rotação da tartaruga em apenas 8 direções. Para contornar essa situação possibilitando formas mais arredondadas, é recomendado reduzir o tamanho das imagens dos blocos semelhantes ao que ocorre em desenhos na forma de *pixel art*.

A primeira versão do RoboCode em PHP desenvolvida sem a compilação condicional resultou em um código com 19.641 linhas, na qual apenas 1.194 foram desenvolvidas pelo programador. A segunda versão em Java utilizando o CIDE e a compilação condicional possibilitou um código com apenas 3.803 linhas, todavia 1457 linhas foram implementadas pelo desenvolvedor. Desta forma, é possível observar que o uso da compilação condicional reduz consideravelmente o número de linhas do projeto final, deixando apenas os conteúdos relevantes para aquele código, todavia a compilação condicional é um processo que ocorre antes do desenvolvimento da aplicação final, de forma que ela seja apenas aplicada nos códigos fornecidos pela LPS, não interferindo no código que o desenvolvedor precisa criar, não reduzindo o seu esforço. Porém, o uso da compilação condicional pode reduzir a CC dos códigos fornecidos pela LPS, como ocorre no script responsável por validação de formulários, que apresenta uma CC de 19 pontos sem a compilação condicional e apenas 5 pontos com o uso da compilação condicional.

Ao avaliar o CIDE como uma ferramenta de suporte ao desenvolvimento de LPS, foi possível ter resultados agradáveis devido ao fato que não há a necessidade de se utilizar as tags da compilação condicional. Todo o trabalho de compilação condicional na ferramenta CIDE é realizado selecionando trechos de códigos ou arquivos e informando a qual determinada *feature* esses dados pertencem. Ao pedir para gerar a estrutura que será usada na aplicação final, basta apenas escolher quais *features* o jogo que será desenvolvido vai usar e a ferramenta se encarrega de remover tudo o que não for utilizado. Entretanto, durante esse processo de remoção de trechos de código, a ferramenta CIDE apresentou algumas falhas ao não remover corretamente trechos de códigos como as chaves, sendo necessário remover manualmente na aplicação final. Numa visão de ensino e pesquisa, a ferramenta CIDE se mostrou bastante rica e promissora, porém numa visão mercadológica, apresentou uma alta dependência da LPS desenvolvida na ferramenta, devido à falta de interoperabilidade entre outras ferramentas.

9 | CONCLUSÃO

O uso de jogos na educação tem se tornando uma ferramenta utilizada para atrair a atenção e motivação aos estudos. Os jogos construcionistas em específicos buscam a motivação dos estudantes e um maior aprofundamento nos conteúdos estudados através da construção de artefatos. Entretanto, o processo de desenvolvimento desses jogos pode se tornar complicado devido ao tempo e custo investido, além de poucos recursos voltados a esse gênero de jogos. Neste artigo é apresentada uma LPS com a finalidade de contribuir com o desenvolvimento de novos jogos construcionistas. Com o desenvolvimento da LPS proposta nesse trabalho, foi possível realizar a construção de novas versões de 3 jogos construcionistas usando a LPS. Durante o desenvolvimento dos jogos foi observado melhorias na redução da CC, redução de número de linhas desenvolvidas pelo programador, menor tempo investido, facilidade no desenvolvimento e uma comparação de um jogo usando a técnica de compilação condicional com a ferramenta de suporte CIDE e sem o uso desta técnica. Para trabalhos futuros pretende-se realizar melhorias nos componentes da LPS e criar um ambiente para disponibilizar toda documentação e códigos.

REFERENCIA

BLANES, D.; INSFRAN, E. A comparative study on model-driven requirements engineering for software product lines. **Revista de Sistemas e Computação**, v. 2, n. 1, p. 3–13, 2012.

DALMON, D. L.; BRANDÃO, L. D. O. Sobre o Desenvolvimento de Software Educacional: proposta de uma Linha de Produto de Software para Módulos de Aprendizagem Interativa. **Revista Brasileira de Informática na Educação**, v. 21, n. 3, p. 113–130, 2013.

FEIGENSPAN, J. et al. Visual support for understanding product lines. **IEEE International Conference on Program Comprehension**, p. 34–35, 2010.

FEIN, G. G. et al. Computer Space: A conceptual and Developmental Analysis of LOGO. In: PSYCHOLOGY PRESS (Ed.). **Constructivism in the Computer Age**. (s.l: s.n.).

GACEK, C.; ANASTASOPOULES, M. Implementing product line variabilities. **Proceedings of the 2001 symposium on Software reusability: putting software reuse in context**, v. 26, n. 3, p. 109–117, 2001. KAFAI, Y. B. Cognitive apprenticeship. In: **The Cambridge Handbook of The Learning Sciences**. (s.l: s.n.). p. 35–46.

KAKOLA, T.; LEITNER, A. Introduction to Software Product Lines: Engineering, Services, and Management Minitrack. **2014 47th Hawaii International Conference on System Sciences**, p. 4984–4984, 2014.

LEBOW, D. Constructivist values for instructional systems design: Five principles toward a new mindset. **Educational Technology Research and Development: ETR&D**, v. 41, n. 3, p. 104–116, 1993.

LESSA FILHO, C. A. C. et al. Um jogo digital baseado no construcionismo. **Revista Brasileira de**

Informática na Educação, v. 23, n. 2, p. 175, 2015.

MARLOWE, B. A.; CANESTRARI, A. S. **Educational Psychology in Context Reading for future Teachers**. Londres: Sage Publications, 2006.

MORIN, B. et al. A generic weaver for supporting product lines. **Proceedings of the 13th international workshop on Software architectures and mobility - EA '08**, p. 11, 2008.

PAPERT, S. **Constructionism**. Disponível em: <<http://www.papert.org/articles/freire/freirePart1.html>>.

PAPERT, S.; HAREL, I. **Constructionism**. Ablex Publishing Corporation, 1991.

POHL, K.; BÖCKLE, G.; LINDEN, F. VAN DER. **Software product line engineering**. Alemanha: Springer, 2005.

REZENDE, F. G. C. et al. Jogo eletrônico e sua influência nas emoções do usuário: Uma análise sobre como os jogos podem estimular emoções relacionadas à aprendizagem. n. Cbie, p. 265–274, 2013.

ROBOCODE. **Robocode**. Disponível em: <<http://robocode.sourceforge.net/>>.

SANTOS, W. O. et al. **Avaliação de Jogos Educativos: Uma Abordagem no Ensino de Matemática**. 26 out. 2015 Disponível em: <<http://br-ie.org/pub/index.php/sbie/article/view/5334>>.

TAROUCO, L. M. R.; KONRATH, M. L. P.; ROLAND, L. C. O professor como desenvolvedor de seus próprios jogos educacionais : até onde isso é possível ? **SBIE - Simpósio Brasileiro de Informática na Educação**, p. 25–28, 2004.

TURI, A. et al. From Fordism To Lean Management : main shifts in automotive industry evolution within the last century. **MakeLearn and TIIM Joint International Conference**, p. 477–485, 2015.

WATSON, A. H.; MCCABE, T. J.; WALLACE, D. R. Structured Testing: A Testing Methodology Using the Cyclomatic Complexity Metric. **NIST Special Publication**, v. 500, n. 235, p. 1–114, 1996.

YOON, D. M.; KIM, K. J. Challenges and Opportunities in Game Artificial Intelligence Education Using Angry Birds. **IEEE Access**, v. 3, p. 793–804, 2015.

METODOLOGIAS GAMIFICADAS PARA A EDUCAÇÃO: UMA REVISÃO SISTEMÁTICA

André Luiz de Souza Brito

Universidade Federal do Rio Grande do Norte
Natal - RN

Charles Andryê Galvão Madeira

Universidade Federal do Rio Grande do Norte
Natal - RN

RESUMO A gamificação é uma abordagem promissora para ajudar a resolver o problema motivacional dos alunos no contexto educacional, através do uso de elementos de jogos que promovem um maior engajamento no processo de ensino-aprendizagem. Muitos trabalhos de pesquisa têm proposto metodologias gamificadas para a educação. No entanto, pouco tem sido explicado sobre a concepção destas metodologias e sobre como tem sido feita a escolha dos elementos de jogos utilizados na solução para a problemática identificada. Esse trabalho propõe uma revisão sistemática da literatura com intuito de observar como ocorre o processo de concepção dos ambientes educacionais gamificados, desde a escolha dos elementos de jogos até as ferramentas utilizadas.

PALAVRAS-CHAVE Gamificação na Educação; Metodologia; Elementos de Jogos.

ABSTRACT Gamification has been a promising approach to tackle student motivation issues in the educational context, through the use of game elements which promote better engagement in the learning process. A lot of research has focused on proposing gamified methodologies for education, but little has been explained about the design and choices made in the construction of the solutions. This work presents a literature review of gamification methodologies with the purpose of observing their creation process, evaluating how and why game elements and tools are usually chosen.

KEYWORDS Gamification in Education; Methodology; Game Elements.

1 | INTRODUÇÃO

A desmotivação estudantil é um fenômeno mundial observado em diferentes níveis de escolaridade (HARTNETT *et al.*, 2014), podendo ser considerada uma consequência da inadequação das práticas tradicionais de ensino frente aos novos aspectos culturais de gerações mais jovens, principalmente no tocante ao acesso à informação e uso ubíquo das tecnologias como ferramentas do dia-a-dia. Isso demonstra que os jovens alunos

necessitam de novos modelos de ensino-aprendizagem que sejam mais interativos, estimulem a proatividade, colaboração, inovação e outras habilidades consideradas essenciais para o novo perfil de formação exigido pela sociedade atual (COUTINHO, LISBÔA, 2011).

Dentre os esforços empreendidos para ajudar a solucionar este problema, uma área que se destaca é a Gamificação (DETERDING *et al.*, 2011), que utiliza elementos e técnicas dos jogos digitais para promover um maior engajamento no processo de ensino-aprendizagem. Muitos trabalhos de pesquisa têm sido direcionados neste sentido, visando propor metodologias gamificadas com aplicação direta em sala de aula ou através de ambientes virtuais de ensino. No entanto, pouco tem sido explicado sobre a concepção destas metodologias e sobre a escolha dos elementos de jogos utilizados como forma de solução para a problemática identificada. Atualmente, momento em que a gamificação encontra-se no vale da desilusão no *Hype Cycle* das tecnologias emergentes em educação da Gartner (DE BRUYCKERE, 2017), as escolhas dos elementos de jogos utilizados nas soluções gamificadas aparentam ter sido feitas, na grande maioria dos casos, de forma aleatória. Esse modo *ad-hoc* de escolha dos elementos utilizados pode ter conexão direta com a obtenção de resultados indesejados a médio e longo prazo na maioria das soluções propostas nos últimos anos (FOGEL, 2015), uma vez que não há garantia que a metodologia contribua de forma eficaz no processo de resolução do problema. Essa também é uma evidência de que o processo de design para a concepção das soluções ainda não é adequado e precisa ser reforçado para garantir a construção de soluções efetivas.

Recentemente, Ogawa *et al.* (2016) apresentaram uma revisão sistemática da literatura para identificar como é realizado o processo de avaliação dos ambientes educacionais gamificados. O presente trabalho, diferentemente do de Ogawa, propõe uma revisão sistemática da literatura com intuito de observar como ocorre o processo de concepção dos ambientes educacionais gamificados, o que engloba os elementos de jogos e as ferramentas utilizados. Com esse propósito, o trabalho é estruturado da seguinte forma: a seção 2 apresenta a metodologia de revisão sistemática aplicada, a seção 3 apresenta uma análise sobre os resultados obtidos do processo de revisão e a seção 4 apresenta as conclusões sobre os achados da pesquisa.

2 | METODOLOGIA DE REVISÃO SISTEMÁTICA

A metodologia de revisão sistemática executada foi baseada no trabalho de Petersen *et al.* (2008) e tem como principal objetivo explorar o estado da arte na área da gamificação na educação, de forma que se tenha um panorama dos principais trabalhos que compõem a vanguarda da área. O processo é composto por diversas etapas: definição das hipóteses de pesquisa, construção do argumento para busca

de trabalhos nos principais repositórios científicos e seleção dos trabalhos relevantes, tendo como produto final uma base que representa uma parcela significativa dos esforços em pesquisa na área.

2.1 Definição das questões de pesquisa

O primeiro passo foi a definição dos questionamentos que nortearam o processo de análise dos resultados da revisão sistemática. Como o interesse é entender a concepção dos modelos gamificados para trabalhar a motivação estudantil, as seguintes questões foram levantadas:

- Q1: Como é o processo de design da solução? Existe uma investigação prévia do cenário?
- Q2: Que elementos de jogos estão sendo mais usados nos modelos de gamificação para educação?
- Q3: Quais as ferramentas mais utilizadas para dar suporte à gamificação criada?
- Q4: Como a motivação é tratada pela gamificação?

As respostas para esses questionamentos permitirão nos dar uma visão geral do que os pesquisadores pensam antes de conceber uma metodologia gamificada e aplicá-la em um cenário real.

2.2 Construção do argumento de busca

Para responder aos questionamentos da seção anterior, foi construído um argumento de busca genérico o suficiente para trazer a maior variedade possível de trabalhos contendo metodologias gamificadas voltadas para a educação:

gamif* and (education or learning) and “student motivation”

Esse argumento foi utilizado para investigar um conjunto de repositórios científicos, a saber: *Scopus*, *ACM Digital Library*, *IEEEExplore*, *Web of Science* e *SpringerLink*. Foi necessário realizar uma adaptação na busca para a base de dados da ACM, visto que na época da busca não era possível a utilização dos caracteres de expressão regular simbolizando “coringas” (*, ?), e a expressão de busca foi construída de forma mais explícita (por exemplo, colocando os termos **gamification** OU **gamified** OU **gamifying**, ao invés de **gamif***). O resultado dessa busca inicial foi um total de 690 trabalhos, datados a partir do ano de 2011.

A partir desse conjunto inicial não refinado, deu-se início ao processo de seleção dos trabalhos mais relevantes mediante os objetivos do estudo proposto neste trabalho.

2.3 Seleção de artigos relevantes para a pesquisa

Para realizar um primeiro filtro do conjunto inicial de trabalhos selecionados, foram adotados os seguintes critérios:

- **Critério 1:** Eliminação de trabalhos sem disponibilidade do arquivo para leitura completa (download ou online);
- **Critério 2:** Eliminação de trabalhos duplicados, ou seja, trabalhos que foram encontrados em diferentes bases;
- **Critério 3:** Eliminação de trabalhos distintos de um mesmo projeto de pesquisa, ou seja, trabalhos diferentes que faziam parte de estudo de longa duração e relatavam momentos ou segmentos diferentes da pesquisa. Nesses casos, deu-se preferência pela versão mais atual e completa do trabalho, também priorizando publicações em periódicos sobre publicação em eventos;
- **Critério 4:** Eliminação de trabalhos com data de publicação anterior ao ano de 2012, por observar nas bases que a partir desta data que se intensificaram as publicações de trabalhos relacionados à gamificação;
- **Critério 5:** Eliminação de artigos que abordavam a gamificação em áreas não relacionadas à educação;
- **Critério 6:** Eliminação de artigos que abordavam técnicas de motivação e engajamento distintas da gamificação (game-based learning, por exemplo).

Para a eliminação via critérios 3, 5 e 6 foi efetuada uma leitura dos *abstracts* e resumos disponíveis para detectar se o artigo atendia às condições aqui pré-definidas. A partir da leitura, foi possível perceber que ainda existe muita confusão com relação à terminologia da área, com um número considerável de trabalhos apresentando propostas de aprendizado baseado em jogos como um modelo de gamificação, o que indica que ainda existe necessidade de conscientização sobre a diferença entre as duas metodologias.

Após esse primeiro filtro, restaram 32 trabalhos, para os quais foi feita uma leitura mais aprofundada da seção de contexto do trabalho e da metodologia aplicada. Nesse ponto do processo já é possível observar o panorama da pesquisa na área da gamificação nos últimos cinco anos. Observa-se um aumento no número de publicações em eventos e periódicos sobre o tema gamificação, como visto no gráfico da Figura 1, em especial com um aumento significativo de publicações em periódicos nos últimos dois anos.

Outro ponto interessante de observar é com relação aos tipos de estudo executados. Para efeito de classificação, os trabalhos foram agrupados em quatro categorias:

- **Revisão da Literatura:** trabalhos que traziam uma investigação da literatura concernente ao tema de gamificação;

- **Propostas de Modelo Gamificado:** trabalhos que apresentavam uma proposta de modelo gamificado, seja um framework conceitual ou uma proposta de ferramenta;
- **Estudos de Caso:** trabalhos que realizaram experimentação científica através de estudos de caso, seja via o uso de um modelo ou ferramenta própria, ou através do uso de tecnologias já existentes;
- **Artigos de Opinião:** trabalhos que traziam ponto de vista dos autores sobre o tema da gamificação, mas que não contém um método científico de pesquisa explicitado.

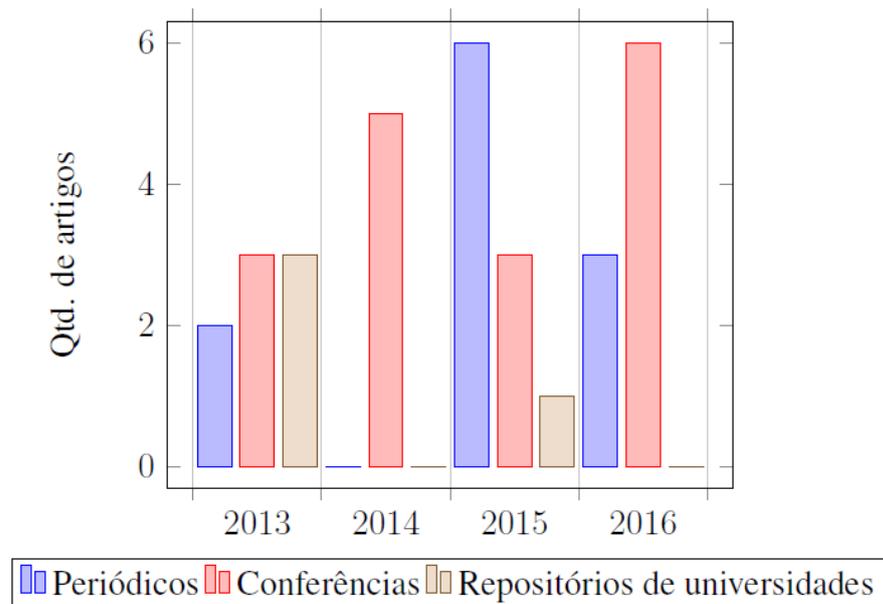


Figura 1. Gráfico contendo a quantidade de artigos analisados que foram publicados a cada ano conforme fontes de publicação.

Percebe-se que existe um interesse crescente em pesquisas através de estudos de caso e observação empírica, o que possivelmente é um resultado do amadurecimento do campo de pesquisa que passa a conduzir investigações com um maior rigor científico em termos de técnicas e metodologias, em detrimento dos trabalhos iniciais que detêm um caráter mais exploratório, de conhecimento e sensibilização para a importância da área. A Tabela 1 mostra os quantitativos para cada tipo de trabalho considerado.

Tipo de Estudo	Total de artigos
Artigos de Opinião	3
Estudos de Caso	17
Proposta de Modelos Gamificados	6
Revisão da Literatura	6

Tabela 1. Quantitativo de trabalhos por tipo de estudo realizado.

Dentre os 17 trabalhos que conduziram experimentos de estudo de caso, houve

uma forte predileção de realizar a avaliação dos resultados através de métodos quantitativos (76%). Uma possível explicação para esse fato pode estar no uso de ferramentas para viabilizar os modelos gamificados, que facilitam a coleta de dados a partir da interação dos usuários, permitindo assim que os pesquisadores fizessem comparações com valores previamente obtidos antes de aplicar a gamificação no mesmo cenário. Apesar disso, é interessante reforçar que a coleta de dados qualitativos pode ter um valor muito grande principalmente no entendimento de como os mecanismos de gamificação são percebidos pelos usuários do processo (IOSUP, EPEMA, 2014), o que é um *feedback* valioso para entender como a abordagem afeta o aprendizado e como ela pode ser modificada para engajar um perfil específico de usuário (BARATA *et al.*, 2015).

Para refinar ainda mais a base de artigos utilizada, aplicou-se um segundo processo de eliminação, retirando do conjunto de artigos aqueles que não apresentavam uma proposta de metodologia gamificada, mas apenas faziam uma investigação sobre a área. Após esse passo, o conjunto de artigos passou a contar com 21 trabalhos, que foram lidos de forma completa e catalogados para montar o referencial do estado da arte em modelos de gamificação aplicados à educação. A Tabela 2 mostra como o conjunto final foi selecionado a partir das bases de pesquisa, fazendo a ressalva que alguns trabalhos podem ser encontrados em mais de uma delas, mas foram contabilizados apenas em uma das bases (não houve um critério maior de agrupamento nesse sentido, apenas sendo contado para a base onde ele foi inicialmente encontrado).

Base de Dados	Total de artigos
ACM Digital Library	6
IEEEExplore	4
ScienceDirect	6
Scopus	4
SpringerLink	1
Total	21

Tabela 2. Artigos por base de dados (refinado).

Com relação aos níveis de ensino nos quais os modelos gamificados foram aplicados, encontramos uma predominância de aplicações no ensino superior, com 62% dos trabalhos atuando com alunos desse nível. Uma possível justificativa é que, devido ao trabalho extra que uma metodologia gamificada produz (IOSUP, EPEMA, 2014), os pesquisadores realizaram os experimentos em suas próprias turmas, dentro da mesma instituição onde o projeto de pesquisa estava sendo executado, ou em instituições parceiras.

3 | ANÁLISE DE RESULTADOS

Com a base final de 21 artigos, foi realizada uma análise para confrontar os dados com os questionamentos de pesquisa levantados no início do processo de revisão sistemática. A discussão em cima do que foi observado é feita nas subseções seguintes.

3.1 Processo de design da solução

O ponto principal da revisão foi entender o processo de construção das soluções. Dentre os trabalhos encontrados, apenas três apresentam uma investigação, pelo menos parcial, do ambiente em que se deseja aplicar a gamificação, como visto na Figura 2. O entendimento da problemática é um ponto fundamental para que a solução gamificada proposta apresente uma maior probabilidade de funcionar de forma adequada. A partir do momento em que esse passo não é executado no processo de design da solução, não há mais como garantir que a gamificação tenha um efeito positivo. Soluções gamificadas concebidas de forma superficial podem, certas vezes, ter um efeito contrário ao esperado, desmotivando os alunos ao longo do processo (ATTALI, ARIELI-ATTALI, 2015; BERKLING, Thomas, 2013).

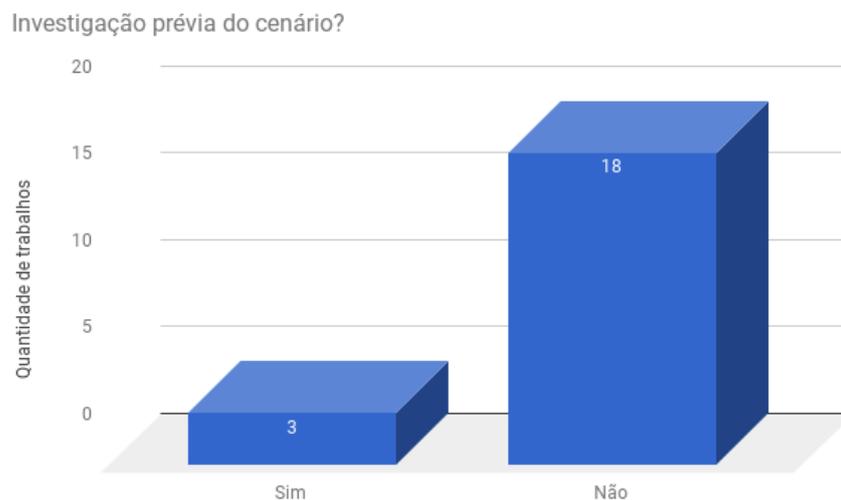


Figura 2. Quantidade de trabalhos que realizaram investigação prévia do cenário antes de propor a gamificação.

Dos 18 trabalhos que não realizaram uma investigação do cenário no qual a solução gamificada foi aplicada, 10 não justificam porque escolheram um determinado conjunto de elementos de jogos para montar a solução, ou como esses elementos podem influenciar na motivação dos alunos. É interessante notar que os outros trabalhos escolheram os elementos de jogos baseados na popularidade do seu uso, apenas seguindo aqueles já amplamente utilizados em pesquisas anteriores (DOMÍNGUEZ *et*

al., 2013; BERKLING, THOMAS, 2013). Esse tipo de conduta não deveria ser adotado pois não garante o sucesso da aplicação da gamificação uma vez que os aspectos motivacionais dos alunos podem ser bem distintos nos diversos cenários investigados pelas pesquisas. Em alguns trabalhos, a gamificação foi projetada com o objetivo de avaliar o efeito de alguns elementos básicos específicos tais quais pontos (ATTALI, ARIELI-ATTALI, 2015), medalhas (DENNY, 2013), (DA ROCHA SEIXAS *et al.*, 2016) e recompensas (HANUS, FOX, 2015). Como o foco dessas pesquisas era a investigação da influência de um único elemento, o modelo de gamificação criado acabou sendo restringido pelos objetivos dos trabalhos.

De uma maneira geral, observa-se que a gamificação tem potencial como técnica para engajar os alunos no ambiente educacional. Porém, o seu processo de concepção deveria ser precedido de uma investigação de como tratar eficazmente os aspectos motivacionais dos alunos, verificando se eles têm interesse neste tipo de solução e estimulando-os para atingir os objetivos de aprendizado. Pelo apurado, poucos trabalhos fazem esse tipo de análise e implantam metodologias gamificadas de uma forma intuitiva e empírica, o que pode ajudar a explicar a existência de sucessos e fracassos em ambientes similares de experimentação, nos permitindo responder ao questionamento **Q1**.

3.2 Elementos de jogos utilizados

A Figura 3 ilustra a frequência de uso de cada um dos elementos de jogos encontrados nas soluções propostas nos artigos selecionados e responde ao questionamento **Q2**. A análise dos dados comprova que existe um favorecimento no uso de elementos típicos do modelo *Points, Badges and Leaderboards* — PBL, baseado em elementos mais comumente associados aos jogos (pontos, medalhas e rankings). Esse modelo costuma ser amplamente adotado por sua facilidade de implementação e ampla divulgação, porém estudos apontam que o uso desses recursos de forma superficial não atinge o efeito de motivação desejado (DE MARCOS *et al.*, 2016). Chou (2015) também destaca que apenas o uso *ad-hoc* de elementos de jogos não garante que uma solução gamificada se torne eficaz. Ao invés disso, ele reforça o ponto de que a gamificação é um processo de design que necessita do entendimento profundo do ambiente e das necessidades do público-alvo.

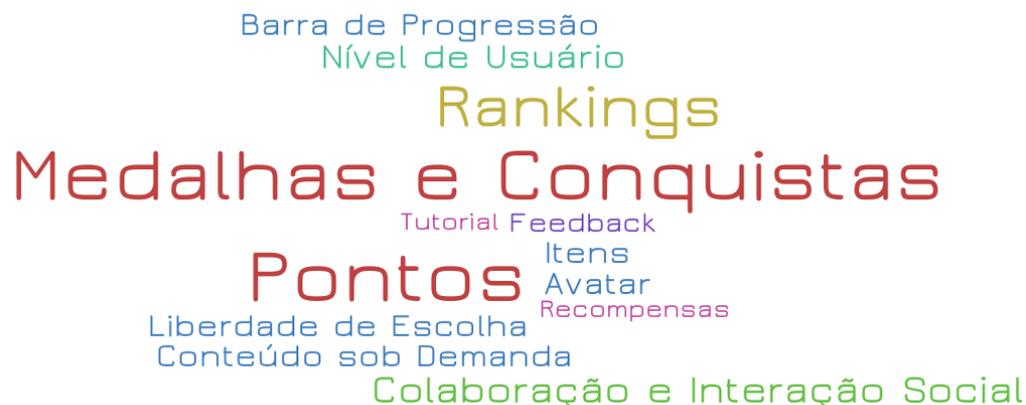


Figura 3. Nuvem de palavras representando a frequência com que cada elemento aparece nas soluções propostas nos artigos.

Os elementos de jogos podem tratar a motivação a partir de dois aspectos distintos: a motivação extrínseca e a motivação intrínseca (HANUS, FOX, 2015). Motivação extrínseca ocorre quando existe um elemento responsável pela motivação, e seu efeito dura apenas enquanto este elemento está disponível para o aluno, como por exemplo pontos, medalhas e recompensas. Já a motivação intrínseca decorre do prazer inerente de realizar a atividade, sem a necessidade de existir um estímulo externo, e normalmente é associada com a liberdade de escolha, criatividade e expressão. O modelo PBL, e a maior parte dos trabalhos analisados, se apoia fortemente em elementos de motivação extrínseca, o que responde ao questionamento **Q4**.

Vale a pena destacar que alguns trabalhos iniciaram a exploração de outros elementos de jogos, como o *feedback* rápido (HOLMAN *et al.*, 2013), sistema de níveis de usuário e progressão no curso (BERKLING, THOMAS, 2013), avatares (KRAUSE *et al.*, 2015) e missões (BARATA *et al.*, 2015), e constituem primeiros passos na busca por elementos de jogos que possam tratar a motivação estudantil em um aspecto de médio e longo prazo de forma mais efetiva.

3.3 Ferramentas utilizadas

No aspecto de ferramentas utilizadas para experimentação com a gamificação observa-se como resposta para a questão **Q3** uma predileção por ferramentas automatizadas, como sistemas de gestão de conhecimento ou softwares específicos para realização da gamificação. Isso possivelmente acontece como uma forma de mitigar o aumento da carga de trabalho com planejamento de atividades e acompanhamento dos alunos, que passa a ocorrer em um nível mais individualizado.

A maior parte dos trabalhos utiliza ferramentas e sistemas já existentes, como o *Moodle* (GENÉ *et al.*, 2014), o *ClassDojo* e *ClassBadges* (DA ROCHA SEIXAS *et al.*, 2016), *Blackboard* (DOMÍNGUEZ *et al.*, 2013) e o *Khan Academy* (RUIPÉREZ-VALIENTE *et al.*, 2016), dentre outros. Um problema que normalmente se encontra no uso de ferramentas prontas é a falta de flexibilidade, já que a gamificação normalmente

fica restrita às possibilidades ofertadas pela ferramenta. Tentando se livrar dessas limitações, alguns trabalhos desenvolveram ferramentas próprias (HOLMAN *et al.*, 2013; DICHEVA *et al.*, 2014), ou adotaram um esquema híbrido, executando parte da gamificação em uma ferramenta própria e parte em uma ferramenta ou ambiente já (DE MARCOS *et al.*, 2016).

É importante refletir sobre o papel da ferramenta na metodologia gamificada. Percebe-se que alguns trabalhos moldaram a gamificação a partir das ferramentas disponíveis, o que pode ser prejudicial à metodologia desenvolvida. Ao invés disso, entende-se que primeiro deve ser traçado o plano de gamificação, e depois os ajustes devem ser feitos de acordo com os recursos disponíveis para a implementação. Em alguns casos, é cabível até mesmo uma avaliação de custo benefício entre criar uma nova ferramenta customizada para o modelo idealizado, ou buscar soluções já prontas dentro da comunidade acadêmica e/ou mercado.

4 | CONCLUSÕES

Este trabalho teve como objetivo investigar como o processo de concepção de metodologias gamificadas aplicadas à educação usualmente ocorre. Para isso, foi efetuada uma revisão sistemática da literatura, a partir da qual apurou-se que a maior parte dos trabalhos não realiza uma investigação prévia do cenário no qual a gamificação é aplicada (**Q1**), escolhendo os elementos de jogos por popularidade de uso ou predileção/intuição dos próprios pesquisadores. Essa falta de um processo apropriado de design da gamificação pode ser um dos fatores que contribuem para a ocorrência de experimentos com resultados inconclusivos ou negativos.

Outro ponto observado é que a maior parte dos trabalhos adotou nas suas soluções elementos de jogos como pontos e medalhas (**Q2**), que estimulam a motivação extrínseca (**Q4**) e cujo efeito de engajamento é temporário. Existe uma necessidade de se explorar elementos de jogos que estimulem a motivação intrínseca, com maior capacidade de promover transformações mais duradouras nos estudantes. Alguns desses elementos, como narrativa e liberdade de escolha do aluno, têm um potencial enorme de aplicação dentro do contexto educacional. Outro ponto de observação foi o conjunto de ferramentas utilizadas, existindo uma predileção pelo uso de ambientes virtuais e ferramentas já existentes para dar suporte à gamificação (**Q3**).

Com o panorama obtido pela pesquisa, o próximo passo será na proposição de um processo de construção de modelos gamificados para a educação. Este processo estabelecerá uma sequência de ações que iniciarão com o entendimento da problemática, execução da gamificação no ambiente proposto e análise dos resultados obtidos para melhoria contínua do modelo criado. Espera-se que a definição desse processo contribua significativamente para construção de metodologias gamificadas

mais eficientes em resolver os reais problemas de motivação dos alunos.

REFERÊNCIAS

- ATTALI, Y.; ARIELI-ATTALI, M. **Gamification in assessment: Do points affect test performance?** *Computers & Education*, 83:57–63, 2015.
- BARATA, G.; GAMA, S.; JORGE, J.; GONÇALVES, D. **Gamification for smarter learning: tales from the trenches.** *Smart Learning Environments*, 2(1):10, 2015.
- BERKLING, K.; THOMAS, C. **Gamification of a software engineering course and a detailed analysis of the factors that lead to it's failure.** In *Interactive Collaborative Learning (ICL)*, 2013 International Conference on Learning, pages 525–530, 2013.
- CHOU, Y.-k. **Actionable gamification: Beyond points, badges, and leaderboards.** Octalysis Media, Fremont, CA, USA, 2015.
- COUTINHO, C. P.; LISBÔA, E. S. **Sociedade da informação, do conhecimento e da aprendizagem: desafios para educação no século XXI.** *Revista de Educação*, 18(1):5– 22, 2011.
- DA ROCHA SEIXAS, L.; GOMES, A. S.; DE MELO FILHO, I. J. **Effectiveness of gamification in the engagement of students.** *Computers in Human Behavior*, 58:48– 63, 2016.
- DE BRUYCKERE, P. **Gartner's hype cycle for education 2016.** Hype Cycle Research Methodology, 2017. <<https://theeconomyofmeaning.com/2016/08/09/gartners-hype-cycle-for-Education-2016/>>. Acesso em: 21 de junho de 2017.
- DE MARCOS, L.; GARCIA-LOPEZ, E.; GARCIA-CABOT, A. **On the effectiveness of game-like and social approaches in learning: Comparing educational gaming, gamification & social networking.** *Computers & Education*, 95:99–113, 2016.
- DENNY, P. **The effect of virtual achievements on student engagement.** In *Proceedings of the SIGCHI conference on human factors in computing systems*, pages 763–772. ACM, 2013.
- DETERDING, S.; DIXON, D.; KHALED, R.; NACKE, L. **From game design elements to gamefulness: defining gamification.** In *Proceedings of the 15th international academic MindTrek conference: Envisioning future media environments*, pages 9–15. ACM, 2011.
- DICHEVA, D. ; IRWIN, K.; DICHEV, C.; TALASIL, S. **A course gamification platform supporting student motivation and engagement.** In *Web and Open Access to Learning (ICWOAL)*, 2014 International Conference on, pages 1–4. IEEE, 2014.
- DOMÍNGUEZ, A.; SAENZ-DE-NAVARRETE, J.; DE-MARCOS, L.; FERNANDEZ-SANZ, L.; PAGÉS, C.; MARTINEZ-HERRAIZ, J.-J. **Gamifying learning experiences: Practical implications and outcomes.** *Computers & Education*, 63:380–392, 2013.
- FOGEL, G. **Will 80% of gamification projects fail? giving credit to gartner's 2012 gamification forecast,** 2015. <<http://www.gameeffective.com/will-80-of-gamification-projects-fail/>>. Acesso em: 21 de junho de 2017.
- GENÉ, O. B.; NÚÑEZ, M. M.; BLANCO, A. F. **Gamification in mooc: challenges, opportunities and proposals for advancing mooc model.** In *Proceedings of the Second International Conference on Technological Ecosystems for Enhancing Multiculturality*, pages 215–220. ACM, 2014.
- HANUS, M. D.; FOX, J. **Assessing the effects of gamification in the classroom: A longitudinal study on intrinsic motivation, social comparison, satisfaction, effort, and academic performance.** *Computers & Education*, 80:152–161, 2015.

- HARTNETT, M.; ST GEORGE, A.; DRON, J. **Exploring motivation in an online context: A case study**. Contemporary Issues in Technology and Teacher Education, 14(1):31–53, 2014.
- HOLMAN, C.; AGUILAR, S.; FISHMAN, B. **Gradecraft: What can we learn from a game-inspired learning management system?** In Proceedings of the Third International Conference on Learning Analytics and Knowledge, pp.260–264. ACM, 2013.
- IOSUP, A.; EPEMA, D. **An experience report on using gamification in technical higher education**. In Proceedings of the 45th ACM technical symposium on Computer science education, pages 27–32. ACM, 2014.
- KRAUSE, M.; MOGALLE, M.; POHL, H.; WILLIAMS, J. J. **A playful game changer: Fostering student retention in online education with social gamification**. In Proceedings of the Second (2015) ACM Conference on Learning@ Scale, pp. 95–102. ACM, 2015.
- OGAWA, A.; KLOCK, A. C. T.; GASPARINI, I. **Avaliação da gamificação na área educacional: um mapeamento sistemático**. In Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação - SBIE), volume 27, pp. 440-449, 2016.
- PETERSEN, K.; FELDT, R.; MUJTABA, S.; MATTSSON, M. **Systematic mapping studies in software engineering**. In EASE, volume 8, pp.68–77, 2008.
- RUIPÉREZ-VALIENTE, J. A.; MUÑOZ-MERINO, P. J.; KLOOS, C. D. **An analysis of the use of badges in an educational experiment**. In Frontiers in Education Conference (FIE), pp.1–8. IEEE, 2016.

APRESENTAÇÃO

No terceiro e último eixo é abordada a aprendizagem de máquina (machine-learning), aplicada a educação e aprendizado. Neste primeiro trabalho o conceito de Estilos de Aprendizagem (EA) da psicologia cognitiva e pedagogia, são propostos em sistemas educacionais adaptativos, porém utilizando do mesmo método que é a de questionários, inviável em algumas situações, a proposta é investigar os algoritmos de classificação relacionados à aprendizagem de máquina para levantar os estilos de aprendizagem de estudantes oriundos da interação com o sistema educacional.

Sistemas adaptativos e inteligentes são utilizados no próximo trabalho para detecção do Estilo de Aprendizagem, como as técnicas de Aprendizagem por Reforço (AR) são consideradas de lenta convergência, foi proposto em seu lugar a um Sistema Adaptativo e Inteligente para a Educação (SAIE) utilizando a técnica de Dynamic Scripting, considerada mais rápida que o AR para reduzir o número de problemas de aprendizagem para EAs estática e dinâmicas reduzindo ainda o número de interações do sistema.

A validação de variáveis de comportamento de autorregulação da aprendizagem é o tema neste último trabalho, investigando a aplicabilidade de modelos de Regressão Múltipla no contexto da EAD, foi proposto uma abordagem híbrida entre questionários e variáveis de arquivos de log objetivando analisar as afirmações dos alunos respondida pelo questionário explicaria o comportamento dentro da plataforma de LMS – Moodle.

Everson Mario Novak

Mestrando em Informática - PUCPR

DETECÇÃO AUTOMÁTICA DE ESTILOS DE APRENDIZAGEM: UMA ANÁLISE COMPARATIVA DE CLASSIFICADORES APLICADOS EM UM CENÁRIO REAL DE APRENDIZADO

Lucas Daniel Ferreira

Universidade de São Paulo
Instituto de Ciências Matemáticas e
Computação
São Carlos – SP

José Fernando Rodrigues Jr

Universidade de São Paulo
Instituto de Ciências Matemáticas e
Computação (ICMC-USP)
São Carlos – SP

RESUMO Diversos estudos nas áreas de psicologia cognitiva e pedagogia apontam que cada indivíduo possui diferentes maneiras de captar, processar, analisar e organizar informações durante o processo de aprendizado, o que fundamenta o conceito de Estilos de Aprendizagem (EA). Em vista disso, diversos sistemas educacionais adaptativos foram propostos com o intuito de proporcionar conteúdo personalizado em seus cursos. Porém, em boa parte dos casos, estes sistemas fazem uso de questionários para identificar os estilos de aprendizagem, e este método pode mostrar-se inviável em algumas situações. Deste modo, a proposta neste trabalho é investigar diferentes

algoritmos relacionados ao aprendizado de máquina (especialmente algoritmos de classificação) aplicados à detecção automática dos estilos de aprendizagem de estudantes, a partir de suas interações em um sistema educacional. Dentre os inúmeros modelos de EA propostos na literatura, optou-se por usar o modelo de Felder-Silverman como base para os experimentos. Para fins de experimentação, foram avaliadas as taxas de acertos e erros dos algoritmos em relação aos resultados apontados pelo questionário ILS, em cada uma das dimensões do modelo de Felder-Silverman. Os resultados apontaram para o uso de mais de um classificador - Naïve Bayes e aprendizagem baseada em instância - dependendo da dimensão do estilo de aprendizagem. A metodologia aplicada foi comparada com sete trabalhos correlatos da literatura; Os resultados demonstraram uma performance superior aos trabalhos anteriores em quase todas as dimensões.

1 | INTRODUÇÃO

Inúmeras instituições de ensino, tais

como colégios e universidades, fazem uso de métodos tecnológicos para fornecer e administrar seus cursos. Devido à popularização do acesso à internet esta prática está em constante crescimento, o que torna o conceito de ensino à distância uma tendência no contexto educacional. Em um ambiente de aprendizado online, são comumente utilizados Sistemas de Gerenciamento do Aprendizado - também conhecidos pela nomenclatura em inglês *Learning Management Systems* (LMS) - como ferramentas de apoio ao ensino; tais ferramentas proporcionam ao docente funcionalidades para criar, gerenciar e compartilhar conteúdo com os estudantes de maneira prática e efetiva. Porém, ao contrário do ensino tradicional (sala de aula) - onde o professor pode facilmente visualizar a maneira segundo a qual cada estudante se comporta - um problema apresentado por esses sistemas é a dificuldade que os professores encontram ao tratar as características individuais de cada aluno no decorrer do curso.

Com efeito, cada pessoa possui habilidades, necessidades, conhecimentos prévios e motivações próprias, e estas características são fundamentais para determinar se um indivíduo encontra mais dificuldades do que outros para aprender em certo modelo de ensino (GRAF, 2007). Além disso, diversos estudos nas áreas de psicologia cognitiva e pedagogia indicam que cada indivíduo possui diferentes maneiras de captar, processar e analisar informações durante o processo de aprendizado, o que fundamenta o conceito de Estilos de Aprendizagem (EA). De maneira geral, assume-se que os EA referem-se aos métodos que um indivíduo aplica com frequência para captar, organizar e interpretar novas informações em situações onde há aquisição de conhecimento. Estes comportamentos representam a maneira segundo a qual um aluno observa o ambiente, recebe informações, as processa, compreende seu significado e então as transforma em conhecimento (JONASSEN & GRABOWSKI, 2012).

A teoria dos Estilos de Aprendizagem tem motivado diversas pesquisas na área da computação aplicada à educação ao longo dos anos, pois, entende-se que a construção e atualização de um modelo de perfil dos estudantes pode melhorar os processos de ensino e aprendizagem de diversas maneiras. Por exemplo, durante a criação do conteúdo de um curso, os professores podem fazer uso dessa informação para compreender mais precisamente a preferência dos estudantes quanto ao material de estudos. Por outro lado, o aluno pode avaliar seu estilo de aprendizagem e identificar quais são seus pontos fortes e fracos durante o aprendizado, o que pode direcionar seus trabalhos a fim de reduzir suas deficiências. E, acima de tudo, a identificação dos EA possibilita a adaptabilidade de conteúdo em cursos oferecidos por meios tecnológicos, o que permite uma melhor experiência de aprendizagem para o aluno, que passa a interagir com recursos e atividades mais adequados às suas preferências em um modelo de ensino personalizado para cada indivíduo.

Usualmente, os estilos de aprendizagem são detectados de maneira colaborativa, ou seja, por meio de questionários respondidos pelos estudantes.

Tais ferramentas indicam os EA com precisão, porém esta abordagem possui

algumas limitações. De acordo com diversos autores (GARCÍA, 2007; GRAF, 2009; DUNG & FLOREA, 2012), este método pode mostrar-se impreciso ou inviável em alguns casos, pois depende diretamente da motivação e autoconhecimento dos estudantes ao preencher os questionários, além de demandar esforço adicional para a obtenção de um modelo, visto que alguns questionários possuem mais de 100 itens. Além disso, esta é uma classificação estática, pois o aluno constrói seu modelo apenas uma vez (geralmente no início do curso).

É hipótese deste trabalho, e de outros existentes (GARCÍA, 2007; GRAF, 2009; DORÇA et al., 2012; DUNG & FLOREA, 2012; LIYANAGE et al., 2014; SENA, 2016) que uma abordagem de detecção automática dos EA mostra-se mais precisa do que a detecção baseada em questionários, pois independe da motivação do aluno ao preencher os formulários, ou do nível de conhecimento do indivíduo a respeito de suas próprias preferências. Além disso, essa proposta isenta o aluno de qualquer esforço adicional para a construção de seu perfil, deixando-o focado apenas nas tarefas de aprendizado. Por fim, a construção do modelo do estudante é feita de maneira contínua, ou seja, enquanto o usuário interagir com o sistema, seu perfil será atualizado, mantendo a classificação válida em caso de mudanças nos Estilos de Aprendizagem.

Assim, o objetivo deste trabalho é explorar diferentes técnicas de aprendizado de máquina supervisionado, relacionadas especificamente à tarefa de classificação, aplicadas à identificação automática de EA. Para isso, será experimentado um conjunto de algoritmos de diferentes paradigmas (com diversas configurações dos dados) para classificar o Estilo de Aprendizagem no qual se enquadra cada aluno com base em seu comportamento durante o uso de um LMS. Para fins de avaliação da pesquisa, serão comparados sistematicamente os resultados dos classificadores quanto à predição dos EA dos alunos, por meio das medidas de acertos e erros obtidas. Ao fim do estudo, espera-se encontrar uma combinação de técnicas e configurações de dados que atinja resultados satisfatórios quando comparada aos trabalhos correlatos no que diz respeito à efetividade da classificação. Além disso, serão levantadas discussões a respeito dos resultados obtidos, a fim de apontar os êxitos e limitações do estudo e contribuir para o cenário atual da modelagem automática de estudantes.

2 | FUNDAMENTAÇÃO TEÓRICA

O termo “Estilos de Aprendizagem” não possui uma definição específica, porém pode ser entendido como uma descrição das ações e comportamentos que determinam as preferências de um estudante durante o processo de aprendizagem. A definição mais comumente aceita é a de Keefe (1979), que considera os EA como “uma composição de características cognitivas, afetivas e fatores fisiológicos que servem como indicadores relativamente estáveis de como um aluno percebe, interage

e responde ao ambiente de aprendizagem”. De maneira geral, os EA referem-se aos métodos que um estudante aplica com frequência para observar o ambiente, captar informações, as processar, compreender seu significado e então transformá-las em conhecimento.

Muitos teóricos educacionais e pesquisadores consideram o EA como um fator importante no processo de ensino e concordam que incorporá-los na educação tem potencial para tornar a aquisição de conhecimento mais fácil para os estudantes (GRAF, 2007). Felder (1988), por exemplo, argumenta que os alunos com uma forte preferência por um EA específico podem ter dificuldades nos estudos caso seu estilo de aprendizagem não seja suportado pelo ambiente de ensino. Por outro lado, pode-se assumir que incorporar os EA dos alunos nos métodos de ensino torna a aprendizagem mais fácil e eficiente.

Inúmeros autores propuseram modelos para definir e classificar Estilos de Aprendizagem capazes de mapear o comportamento de estudantes. Estes modelos são construídos de forma distinta por cada autor, que por sua vez, possui influências e embasamentos em diferentes teorias da psicologia educacional. Neste trabalho, é explorado o Modelo de Estilos de Aprendizagem de Felder e Silverman (FSLSM) (FELDER & SILVERMAN, 1988), que será detalhado a seguir. Uma vasta revisão acerca de modelos de estilos de aprendizagem está disponível em Coffield et al. (2004).

2.1 Modelo de estilos de aprendizagem de Felder e Silverman

O modelo de Felder e Silverman (1988) pode ser compreendido como um compilado de um conjunto de modelos propostos na literatura a fim de dar início a um modelo de maior amplitude e possibilidades. As dimensões dos EA propostos por Felder-Silverman estão relacionadas com as estratégias de captação, percepção, organização e processamento de informações adotadas pelos estudantes durante a aprendizagem. Portanto, Felder e Silverman sintetizaram descobertas de diversos estudos que resultaram em um modelo de estilos de aprendizagem composto por quatro dimensões distintas, que são detalhadas a seguir.

- **Processamento (ativos x reflexivos):** os estudantes ativos mais bem processam a informação enquanto executam alguma atividade, precisam experimentar para compreender, poderão iniciar as tarefas prematuramente e gostam de participar de trabalhos em grupo. Já os estudantes reflexivos precisam compreender para experimentar, demoram a iniciar as atividades e preferem trabalhos individuais.
- **Percepção (sensoriais x intuitivos):** indivíduos intuitivos possuem grande capacidade para distinguir e interpretar símbolos e textos quando comparados aos sensoriais. Além disso, estão propensos a tomar decisões com base na intuição, ou seja, com base em estímulos oriundos do inconsciente. Por outro

lado, os indivíduos sensoriais tomam decisões e percebem as informações baseando-se fortemente nos sentidos (visão, audição, tato, olfato), e tendem a ter maior atividade na região consciente do cérebro.

- **Captação (visuais x verbais):** indivíduos com maior tendência para o polo visual possuem melhor captação de informações que se apresentam de forma gráfica, tais como ilustrações, figuras, diagramas, fluxogramas, esquemas e mapas. Em contrapartida, aqueles mais inclinados ao polo verbal estão relacionados às propriedades de leitura e audição. Eles possuem facilidade para compreender conceitos traduzidos em palavras.
- **Organização (sequenciais x globais):** estudantes sequenciais organizam a informação a partir do detalhamento para compreender informações mais generalizadas. Ou seja, dão forte prioridade para as características específicas de um conteúdo, e não se atentam fortemente ao que é abstrato. Por outro lado, estudantes globais organizam a informação a partir de regras gerais, para que a partir destas cheguem ao nível de detalhe. Deste modo, estes indivíduos tendem a ter uma visão geral do todo, porém podem perder dados importantes que constituem este todo.

Este modelo foi escolhido para os experimentos conduzidos neste trabalho por diversas razões. Uma delas, é o fato de que Felder e Silverman agruparam em um único modelo importantes descobertas de outros autores já consolidados neste campo de investigação, tais como Kolb (1976), Pask (1975) e os Indicadores de Tipos de Myers-Briggs (1985). Outra característica importante deste modelo é a praticidade para se identificar o estilo de um estudante, visto que Felder e Soloman desenvolveram um questionário contendo 44 questões (11 questões para cada dimensão), denominado *Index of Learning Styles (ILS)* (FELDER & SOLOMAN, 2005), que aponta as preferências de cada indivíduo de acordo com o modelo proposto. Por fim, a escolha do modelo de Felder-Silverman justifica-se por este ser um dos modelos mais adotados em trabalhos de modelagem automática de estudantes (FELDMAN, 2015; TRUONG, 2016), além de já ser incorporado a diversos sistemas adaptativos, demonstrando bons resultados quanto à classificação de perfis e recomendação de conteúdo.

3 | TRABALHOS RELACIONADOS

Diversas abordagens foram propostas para identificação automática de Estilos de Aprendizagem em ambientes virtuais de ensino. O trabalho de García et al. (2007), por exemplo, foi um dos precursores no tema e representou a base para vários outros trabalhos que relacionam redes bayesianas e o modelo de estilos de Felder-Silverman

(FSLSM). Neste trabalho os autores aplicaram o Teorema de Bayes e observaram 11 padrões de comportamento em 10 estudantes matriculados em um curso de Inteligência Artificial, oferecido via Web. Como resultado é proposto o sistema SAVER (abreviação para *Software de Asistencia Virtual para Educación Remota*), que, por meio de técnicas de Redes Bayesianas, possibilita o cálculo da preferência dos alunos nas dimensões ativo/reflexivo, sensorial/intuitivo e sequencial/global do FSLSM.

Outro trabalho clássico na área e que influenciou diversas pesquisas, é o de Graf, Kinshuk e Liu (GRAF et al., 2009), o qual propõe uma abordagem de identificação automática de EA em sistemas de gestão do aprendizado a partir das interações dos estudantes nos cursos online. Neste trabalho, os autores exploram técnicas baseadas em regras para traçar a correlação de padrões de comportamento online com os estilos de Felder e Silverman. Para a avaliação, um estudo foi conduzido com 75 alunos, comparando os resultados da abordagem automática com os obtidos pelo questionário ILS, onde alcança-se precisão de 79,33%, 77,33%, 76,67% e 73,33% nas dimensões de processamento, percepção, captação e organização, respectivamente.

Dorça et al. apresentam inúmeras contribuições para o contexto da modelagem automática e dinâmica de estudantes por meio de EA. Em um de seus trabalhos (DORÇA et al., 2011), o foco é a investigação de como (além de construir) atualizar gradualmente e constantemente o modelo dos estudantes, visto que as preferências de cada indivíduo podem mudar de maneira inesperada com o decorrer do tempo. Para a avaliação da proposta, este trabalho considera o modelo FSLSM, e realiza uma simulação computacional do comportamento de estudantes em um sistema educacional. Em outro trabalho, Dorça et al. (2012) realizam uma abordagem semelhante, porém baseada em cadeias de Markov. Neste caso, os autores introduzem uma combinação de Estilos de Aprendizagem para detectar e corrigir o modelo dos estudantes, demonstrando bons resultados.

O trabalho de Dung e Florea (2012) apresenta outra proposta para modelagem automática de estudantes usando o modelo FSLSM, bastante influenciada no trabalho de Graf e Liu (2009). Com uma abordagem baseada em regras de mapeamento, são considerados o número de visitas e o tempo gasto pelos estudantes em cada objeto de aprendizagem (que possuem meta dados descritivos). Como resultado, o método proposto obteve precisão de acertos (em relação ao questionário ILS) de 72,73%, 70,15%, 79,54% e 65,91% nas dimensões de processamento, percepção, captação e organização, respectivamente.

Em um trabalho mais recente, Liyanage et al. (2014) propõem um arcabouço para sistemas adaptativos dividido em três módulos. O primeiro, aplica o já conhecido questionário ILS para indicar os EA dos estudantes segundo o modelo de Felder-Silverman. No segundo módulo ocorre a recomendação de objetos compatíveis com cada estilo de aprendizagem; e por fim, no terceiro módulo são disponibilizadas ferramentas para os docentes calibrarem os mecanismos de adaptabilidade. Os experimentos para detecção dos EA foram conduzidos em um grupo de 80 estudantes matriculados no

Moodle, e alcançaram precisões de 65%, 75%, 76,25% e 77,50% para as dimensões de processamento, percepção, captação e organização, respectivamente.

Dentre os trabalhos contemporâneos, Sena et al. (2016) propõem uma abordagem probabilística que combina o modelo FSLSM com as técnicas de inferência probabilística dos Modelos Ocultos de Markov (HMM) a fim de realizar o processo de inferência da preferência do aluno por um determinado estilo de aprendizagem. Para isso, matrizes de emissão e transição de estados do modelo de Markov foram criadas por meio do algoritmo de Viterbi. Para avaliar a proposta, os autores realizam uma simulação do comportamento de estudantes em sistemas educacionais, e alcançam bons resultados quanto à detecção de seus EA.

Conforme apontado por Truong (2016), - que apresenta uma revisão a respeito de desenvolvimentos, problemas e oportunidades na área de EA integrados a Sistemas Educacionais Adaptativos - apesar da variedade de trabalhos conduzidos na área, ainda existem questões em aberto e diversas lacunas a serem investigadas. Uma das questões discutidas é que embora muitos algoritmos de classificação tenham sido explorados, entre os quais Redes Bayesianas e Rules-based são os mais amplamente utilizados, apenas uma pequena proporção de trabalhos considera uma análise comparativa de métodos de classificação neste contexto. Portanto, o presente trabalho diferencia-se dos demais apresentados neste capítulo por atender à oportunidade de aplicar diferentes métodos de classificação ao mesmo problema, com o intuito de comparar desempenhos e levantar discussões a respeito dos resultados obtidos.

4 | MATERIAIS E MÉTODOS

4.1 Contexto, participantes e características do curso

Para a realização dos estudos propostos neste trabalho, foi avaliado um curso de pós-graduação em Reabilitação Auditiva, na área de Fonoaudiologia, oferecido pela Faculdade de Odontologia de Bauru em parceria com o Hospital Samaritano de São Paulo. O curso em questão possui duração de 18 meses, é ministrado integralmente com auxílio do sistema Moodle, e conta com 105 alunos matriculados, oriundos de diversas localidades do Brasil. Dada a separação geográfica, estes estudantes interagem entre si diariamente por meio dos fóruns de discussões e ferramentas de comunicação disponíveis no ambiente virtual. Como parte dos experimentos, os alunos foram solicitados a responder ao questionário ILS no início do curso com o intuito de registrar suas respectivas preferências nas dimensões de processamento, percepção, captação e organização do modelo de estilos de aprendizagem de Felder-Silverman (FSLSM). Os resultados do questionário são apresentados na Tabela 1, a qual indica o percentual de preferência dos estudantes para cada dimensão.

Processamento		Captação		Organização		Percepção	
Ativos	Reflexivos	Visuais	Verbais	Sequenciais	Globais	Sensoriais	Intuitivos
79,38%	20,62%	53,61%	46,39%	69,07%	30,93%	87,63%	12,37%

Tabela 1 - Resultados do questionário ILS.

Os registros de uso do Moodle foram coletados entre setembro de 2015 e agosto de 2016. Ao todo, foram avaliadas as interações dos estudantes com 252 recursos educacionais disponibilizados neste período, o que corresponde ao material didático referente aos 16 primeiros capítulos do curso.

4.2 Recursos educacionais

O curso em questão oferece uma grande diversidade de recursos educacionais aos estudantes, que variam suas características de acordo com o propósito do objeto de aprendizagem e sua forma de apresentação. Na Figura 1, é possível observar a distribuição dos recursos educacionais disponíveis no curso avaliado de acordo com suas características. Conforme pode-se observar, objetos relacionados a materiais didáticos textuais e vídeo-aulas são predominantes (70 e 54 objetos respectivamente). Neste trabalho e em outros semelhantes, assume-se que a distinção dos materiais de acordo com o formato de apresentação (vídeo ou texto) é fundamental para o processo de classificação de estilos de aprendizagem segundo o modelo de Felder-Silverman. Isso se dá, principalmente, devido à dimensão de captação desse modelo, que diferencia os estudantes entre visuais ou verbais.

Além disso, destaca-se o alto número de fóruns de discussões disponíveis no curso (total de 44). Isso ocorre principalmente pela separação geográfica dos participantes, que usaram essa ferramenta como mecanismo central de comunicação. Conseqüentemente, uma alta massa de dados foi gerada decorrente das interações em fóruns de discussões, como por exemplo a criação de tópicos, visualizações, respostas e postagens. Tais interações foram registradas diariamente, o que caracteriza uma alta comunicação entre todos os participantes, que foram fortemente incentivados a usar a ferramenta.

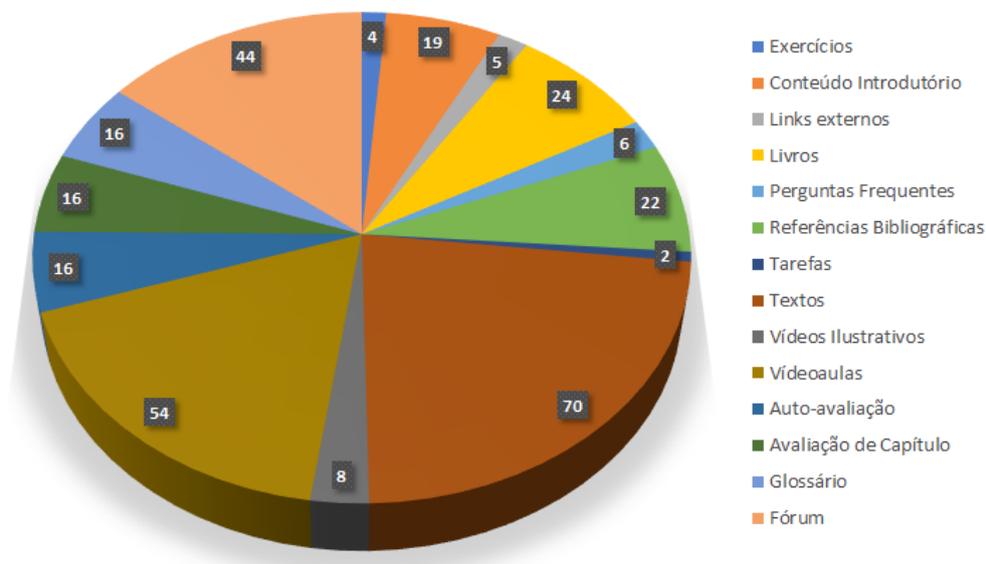


Figura 1 - Recursos educacionais disponíveis no curso avaliado

Adicionalmente, livros, materiais introdutórios e páginas de referências bibliográficas também são recursos recorrentes durante o curso. Os livros possuem objetivo semelhante aos materiais textuais citados anteriormente, porém divergem quanto à forma de apresentação. Neste caso, o estudante pode navegar pelo conteúdo de maneira ordenada e sequencial. Por hipótese, estudantes com predileção por este tipo de recurso podem estar mais inclinados ao estilo de aprendizagem sequencial do modelo de Felder-Silverman, enquanto aqueles com preferência por textos convencionais podem se identificar como globais.

Ademais, objetos relacionados a formulários de auto avaliação, avaliação dos capítulos e glossários acompanham a quantidade de capítulos observados. Em conclusão, objetos relacionados a vídeos ilustrativos, tarefas, exercícios, links externos e páginas de perguntas frequentes representam uma pequena fração dos recursos educacionais, mas ainda assim foram considerados.

4.3 Padrões de comportamento

O modelo de estilos de aprendizagem de Felder e Silverman, assim como os demais modelos propostos na literatura, foi construído com base em conceitos do ensino tradicional. Portanto, para que seja possível detectar esses estilos de maneira automática e dinâmica em um sistema educativo, é necessário mapear o comportamento dos estudantes em ambientes tradicionais para as ações realizadas no ambiente virtual de ensino. Em vista disso, nesta seção são apresentadas as ações desempenhadas pelos usuários no ambiente virtual que serviram como base para o processo de caracterização de perfis durante a execução do curso. Para isso, foi explorado um conjunto de interações e comportamentos capazes de descrever as preferências de cada estudante durante o processo de aprendizagem em um sistema educacional.

Estes padrões de comportamento foram bastante influenciados pelos trabalhos de Graf (2009), García et al. (2007) e Dung e Florea (2012), os quais trabalham com mapeamentos de padrões de utilização dos usuários no sistema Moodle para o modelo FSLSM. Mapeamentos que, por sua vez, foram fortemente baseados no trabalho de Felder e Soloman (2005), os quais apontam as preferências dos estudantes durante o aprendizado e como elas estão relacionadas a cada dimensão do modelo proposto.

Categoria	Recursos	Dados Coletados
Materiais Didáticos	Conteúdo Introdutório	Visitas; Tempo gasto.
	Livro	Visitas; Tempo gasto; Navegação.
	Conteúdo Textual	Visitas; Tempo gasto.
	Conteúdo em Vídeo	Visitas; Tempo gasto.
Atividades	Tarefas	Visitas; Tempo gasto.
	Exercícios	Visitas; Tempo gasto.
Materiais de Apoio	Perguntas Frequentes	Visitas; Tempo gasto.
	Referências Bibliográficas	Visitas; Tempo gasto.
	Links Externos	Visitas; Tempo gasto.
Interação	Fórum de Discussões	Visitas ao fórum; Visitas aos tópicos; Criação de tópicos; Pesquisas; Postagens; Média de palavras; Tempo gasto.
	Glossário	Visitas; Tempo gasto; Postagens.

Avaliação	Auto Avaliação	Visitas; Tempo gasto; Questões respondidas.
	Avaliação do Capítulo	Visitas; Tempo gasto; Questões respondidas.

Tabela 2 - Padrões de Comportamento Avaliados.

Um dos requisitos nesta tarefa é que os padrões de comportamento investigados sejam relevantes o suficiente para a predição automática de EA, ou seja, os dados observados devem ser representativos o bastante para que os algoritmos de classificação baseados em aprendizado de máquina sejam capazes de inferir modelos eficientes e precisos.

A partir das premissas citadas, foi selecionado um conjunto dos registros de interação a serem incorporados a esta experimentação, descritos na Tabela 2. Os objetos referentes ao material didático representam a apresentação do assunto em questão, o que pode ser realizada de diferentes formas (textos em *pdf*, apresentações de slides, vídeo-aulas, livros, etc.). Neste caso, os dados de interação dos usuários levados em conta foram a quantidade de visitas e o tempo médio gasto em acessos para esta categoria de recursos. Além disso, nos objetos apresentados no formato “livro”, foram quantificadas as alterações de navegação entre as páginas, ou seja, o número de trocas de páginas. As mesmas informações foram coletadas quanto aos acessos aos materiais introdutórios, exemplos, vídeos ilustrativos, vídeo aulas, páginas de perguntas frequentes, referências bibliográficas, e exercícios publicados pelos docentes.

Para os formulários de avaliação foram consideradas informações mais detalhadas: o número de questões respondidas, o desempenho do aluno em suas respostas e o tempo gasto durante o teste. Quanto às atividades do glossário, foram coletadas a quantidade de acessos às páginas de glossário, o tempo gasto e a quantidade de publicações de cada aluno. Por fim, para os recursos de comunicação foram considerados o número de visitas ao fórum de discussões, número de visitas aos tópicos do fórum, número de tópicos criados por cada usuário, quantidade de postagens publicadas pelo usuário, tempo médio gasto em recursos do fórum, número de pesquisas realizadas e quantidade média de palavras por publicação.

4.4 Classificação de dados

De maneira geral, o aprendizado de máquina é um processo que permite a descoberta de conhecimento a partir da experiência. Nos casos de aprendizado supervisionado, este processo consiste na atribuição de rótulos aos objetos que

serão usados para treinar o método de aprendizado computacional. O rótulo atribuído corresponde à resposta certa conhecida, a resposta com a qual o método deverá aprender e prever para outros objetos que não aqueles no conjunto de treino. No contexto das tarefas de aprendizado de máquina, o processo de classificação se dá por meio de um algoritmo cujo objetivo é prever o rótulo de novas entradas com base na experiência adquirida a partir de exemplos de entrada já rotulados. Neste caso, o primeiro passo para a classificação é mapear e modelar o conjunto de dados do domínio ao qual pretende-se aplicar o algoritmo, preparando os exemplos para o processamento.

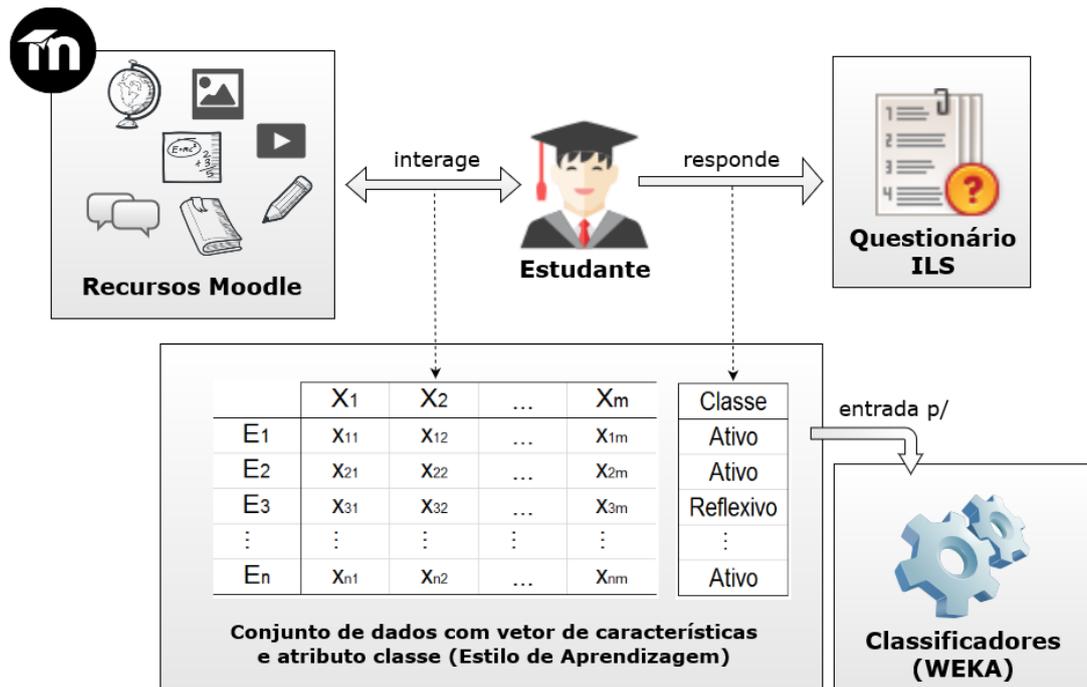


Figura 2 – Metodologia para classificação de Estilos de Aprendizagem

Sendo assim, idealmente o conjunto de dados de entrada, também denominado conjunto de treino, é composto por uma coleção de exemplos, sendo que cada exemplo é constituído por uma lista de atributos independentes, capazes de descrever as características deste exemplo. Portanto, é fundamental para o processo de classificação que essa lista de atributos seja composta por informações significativas e relevantes sobre o exemplo, de modo a representar um objeto do mundo real de maneira precisa. Há também um atributo classe, como já mencionado, o qual é o atributo alvo para os problemas de classificação e regressão.

A ideia na etapa de treinamento dos classificadores é utilizar meios computacionais combinados com modelos matemáticos, estatísticos, associativos ou lógicos para encontrar um modelo que correlacione as características de um exemplo com seu rótulo. De modo geral, o objetivo neste tipo de aplicação é encontrar modelos precisos o suficiente para realizar a predição do rótulo de novos exemplos, usando apenas seu conjunto de atributos, com um alto grau de confiabilidade. Por fim, é realizada a fase

de teste do classificador. Também definida como fase de validação, esta etapa do processo busca avaliar a eficácia do modelo gerado a partir da classificação de novos exemplos ainda não rotulados que compõem o conjunto de teste.

4.5 Metodologia para experimentação

Dadas as definições do processo de classificação de dados, uma das etapas fundamentais foi a construção do conjunto de dados para a execução dos experimentos, onde cada estudante matriculado no curso corresponde a um exemplo observável. Portanto, para cada aluno da amostra, foi coletado um conjunto de atributos que descrevem suas características e preferências durante a experiência com o sistema Moodle. Além disso, cada estudante deveria possuir um rótulo, fundamental para as fases de treinamento e teste das técnicas de aprendizado supervisionado investigadas. Conforme ilustrado na Figura 2, os atributos (ou características) que compõem o conjunto de entrada para os algoritmos são constituídos pelas medidas extraídas dos logs de utilização do sistema Moodle. Estas medidas correspondem às interações e padrões de comportamento descritos na Seção 4.3 (Tabela 2). O rótulo para cada instância corresponde ao estilo de aprendizagem identificado por meio do questionário ILS aplicado aos estudantes no início do curso.

Ao todo foram realizadas quatro classificações distintas durante os experimentos: uma para cada dimensão do modelo de Felder-Silverman. Ou seja, os mesmos conjuntos de atributos descritivos dos estudantes foram avaliados em quatro execuções: uma para cada dimensão. Por fim, o resultado dos algoritmos foi avaliado com validação cruzada de 10 execuções (*10-fold cross validation*) no software WEKA, a qual resultou na extração das medidas de precisão, revocação e F-measure de cada técnica investigada.

5 | EXPERIMENTOS E RESULTADOS

Na Tabela 3 são apresentados todos os algoritmos avaliados neste trabalho, provenientes dos diferentes paradigmas do aprendizado de máquina. Como pode-se observar, para o paradigma estatístico foram experimentados os classificadores correspondentes a *Redes Bayesianas*, *Naive Bayes* e *Support Vector Machine*; com relação a redes neurais, foram explorados o *MultiLayer Perceptron* e a implementação do Weka para *Deep Learning* (DL4J); quanto ao aprendizado baseado em Instâncias foram testados o classificador kNN (também denominado *IBk*) e o K-Star; por fim, para o paradigma simbólico foram avaliadas a árvore de decisão de Hoeffding, a árvore J48 (implementação do algoritmo C4), a *Random Forest* e *Random Tree*. Porém, dentre todos os experimentos realizados, serão apresentados nas próximas seções

somente aqueles que obtiveram os melhores resultados para cada paradigma, com o intuito de abstrair o processo de avaliação. Portanto, para o paradigma estatístico, foi selecionado o algoritmo *Naive Bayes*; quanto às redes neurais, foi selecionado o classificador *MultiLayer Perceptron*; para o paradigma baseado em instâncias, foi selecionado o *IBk*; e, por fim, para o paradigma simbólico, foi escolhido o algoritmo *J48*, que implementa árvores de decisões.

Paradigma Estatístico	Redes Neurais	Baseado em Instâncias	Paradigma Simbólico
Redes Bayesianas; Naive Bayes ; Support Vector Machine.	M u l t i L a y e r Perceptron ; Deep Learning.	Classificador kNN ; K-Star.	Hoeffding Tree; J48 ; Random Forest; Random Tree.

Tabela 3 - Algoritmos avaliados para a classificação de EA. Em negrito, os algoritmos com melhor desempenho.

5.1 Avaliação experimental

Para os experimentos conduzidos neste trabalho, o conjunto de entrada passou por um pré-processamento envolvendo filtros disponíveis no software WEKA. Primeiramente, foi realizado um filtro de instâncias para balanceamento de classes por meio do método *ClassBalancer*. Neste filtro, são atribuídos pesos às instâncias do conjunto de dados para que cada classe tenha o mesmo peso total, mas de maneira que a soma total de pesos em todas as instâncias seja mantida. Esta prática serve como uma solução para problemas de classificação em conjuntos de dados com classes desbalanceadas. Nos experimentos, este problema ocorreu principalmente durante a avaliação da dimensão de percepção do modelo de Felder-Silverman, visto que mais de 87% dos estudantes foram identificados como ‘*sensoriais*’ e apenas 12% foram identificados como ‘*intuitivos*’.

Em uma abordagem inicial, foram realizados experimentos com pré-processamento limitado apenas a balanceamento de classes. No entanto, esta proposta não alcançou resultados viáveis para um cenário real de modelagem de estudantes, visto que a classificação se deu com precisões máximas de 55,2%, 61,9%, 57,6% e 60,3% para as dimensões de processamento, captação, organização e percepção respectivamente. Tais medidas não são consideradas ideais para a adaptatividade de conteúdo em sistemas educacionais, pois se assemelham a um classificador aleatório, e portanto, não são capazes de prever o estilo de aprendizagem de um estudante com eficácia.

Como primeira alternativa para superar estes resultados, foram avaliadas execuções dos algoritmos acompanhadas de um pré-processamento adicional: o filtro de atributos. Sendo assim, nesta execução dos algoritmos, além do método

ClassBalancer foi avaliada a aplicação de uma seleção de características que, como o próprio nome sugere, tem por objetivo eleger um subconjunto de atributos de entrada, reduzindo a dimensão do conjunto de dados. Com essa redução de dimensão, reduz-se a quantidade de informação redundante e/ou espúria, aumentando assim a precisão dos classificadores; além disso, reduz-se a complexidade dos dados, bem como o tempo de processamento para extrair deles algum conhecimento. Ao todo foram avaliadas as técnicas: *CFS Subset Eval*, *Correlation Attribute Eval*, *Info Gain Attribute Eval*, *Principal Components*, e *Wrapper Subset Eval*, disponíveis na biblioteca. Dentre elas, optou-se pela aplicação do método *WrapperSubsetEval*, o qual demonstrou os melhores resultados quanto à precisão dos algoritmos em relação às demais opções.

Em conclusão, todos os resultados demonstrados a seguir foram obtidos a partir da seguinte configuração: 1) construção do conjunto de entrada a partir da base de dados do sistema Moodle (com rótulos identificados pelo questionário ILS); 2) filtro das instâncias com *ClassBalancer*; 3) seleção de atributos com *WrapperSubsetEval*; 4) execução do algoritmo em validação cruzada de 10 execuções (*10-fold cross validation*); e 5) extração das medidas de Precisão, Revocação (Recall) e F-measure de cada uma das técnicas de aprendizado de máquina avaliadas.

5.2 Resultados

Na Figura 3 são apresentados os índices de precisão dos algoritmos avaliados em conjunto com a técnica de seleção de atributos *WrapperSubsetEval*, aplicados à classificação dos estudantes em cada uma das dimensões do modelo de Felder-Silverman. Conforme pode-se observar, a seleção de atributos impactou significativamente o desempenho da predição, de maneira que este pré-processamento, combinado com o rebalanceamento de classes, mostra-se necessário quando da identificação automática de estilos de aprendizagem. A abordagem de refinamento alcançou precisões máximas de 79,4% (*Naive Bayes*), 76,2% (*IBk*), 76,3% (*Naive Bayes*) e 85,9% (*IBk*) para classificar os estudantes quanto às dimensões de processamento, captação, organização e percepção respectivamente. Tais valores superaram consideravelmente os experimentos iniciais, que alcançaram o máximo de 55,2%, 61,9%, 57,6% e 60,3% de precisão nas mesmas tarefas.

Ademais, os experimentos demonstraram que o aprendizado Bayesiano (*Naive Bayes*) obteve o melhor desempenho nas dimensões de processamento e de organização, ao passo que o aprendizado baseado em instâncias (*IBk*) teve o melhor desempenho nas dimensões de captação e percepção. Individualmente, a técnica *IBk* alcançou a melhor média de precisão somando-se os resultados nas quatro dimensões do modelo de Felder-Silverman, com 77,9%. Por outro lado, o algoritmo *J48* obteve a pior, com média de 67,6% de precisão. Portanto, em caso de escolha da melhor técnica de classificação de EA para este estudo de caso, seria selecionada a abordagem baseada em instâncias. Entretanto, nesta avaliação experimental assume-

se que o uso de um único classificador não é suficiente para todas as dimensões, de modo que o uso combinado dos aprendizados Bayesiano e baseado em instâncias provê uma classificação mais apurada para cenários com esta configuração de dados.

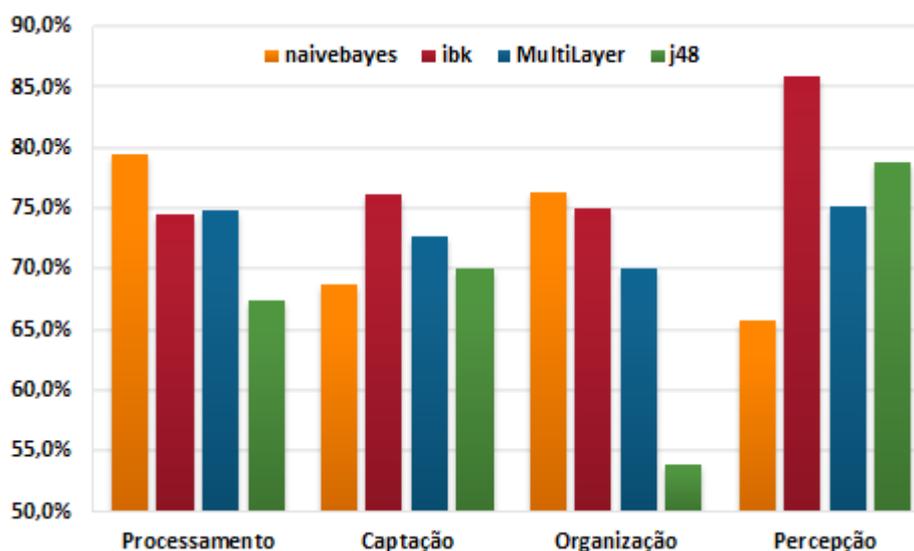


Figura 3 - Análise comparativa da precisão dos classificadores.

No mais, quanto à seleção de atributos, observa-se que os padrões de comportamento mais selecionados pelo algoritmo de poda foram aqueles relacionados à interação com o fórum de discussões, visto que os atributos correspondentes à criação de tópicos, visitas, buscas e publicações no fórum foram selecionados com frequência pelo filtro. Isso pode ser justificado devido ao alto grau de interação com estes recursos por parte dos estudantes que participaram do experimento, produzindo assim, informações mais precisas quanto às suas preferências e comportamento.

5.3 Discussão

Em geral, foram realizadas análises comparativas de diversos algoritmos executados com diferentes configurações, além de uma diversidade de pré-processamentos a fim de se estabelecer um protocolo de experimentação adequado. Como resultado, assume-se que a abordagem proposta alcançou medidas de desempenho viáveis para a aplicação de aprendizado de máquina para a modelagem automática de estudantes em um sistema educacional. Consequentemente, tais resultados validam os padrões de comportamento observados, o estudo de caso conduzido, bem como o processo de experimentação prática dos algoritmos. Quando comparados com trabalhos correlatos, os experimentos conduzidos neste estudo demonstram resultados satisfatórios quanto à precisão dos classificadores.

Conforme apontado na Tabela 4, os percentuais apresentados nesta proposta superam os trabalhos mais clássicos que exploram detecção automática de estilos de aprendizagem, como o de Cha et al. (2006), García et al. (2007), Graf et al. (2009),

e Dung e Florea (2012). Além disso, a precisão obtida encontra-se equiparada aos trabalhos mais recentes (LIYANAGE, 2014; ABDULLAH, 2015), o que sustenta a efetividade da avaliação experimental conduzida.

Autor	Processamento	Captação	Organização	Percepção
Cha et al. (2006)	66,7%	85,7%	71,4%	77,8%
García et al. (2007)	58,0%	---	63,0%	77,0%
Graf et al. (2009)	79,3%	76,7%	73,3%	77,3%
Dung e Florea (2012)	72,7%	79,5%	65,9%	70,1%
Liyanage et al. (2014)	65,0%	76,2%	77,5%	75,0%
Abdullah et al. (2015)	90,0%	76,0%	70,0%	85,0%
Atual proposta	79,4%	76,2%	76,3%	85,9%

Tabela 4 – Comparação da precisão com trabalhos correlatos

Ao comparar os resultados com os trabalhos relacionados, os experimentos demonstraram resultados semelhantes ou superiores em relação à precisão dos outros classificadores em todos os casos. Na dimensão de percepção, o método proposto apresentou o melhor resultado. Comparativamente, o pior desempenho foi na dimensão de processamento, em que o trabalho de Abdullah alcançou 90,0% de precisão contra 82,2%. No entanto, o trabalho de Abdullah considerou apenas 35 alunos, enquanto que o presente trabalho considerou 105 alunos.

Todavia, uma análise comparativa dos índices de precisão alcançados pelos demais trabalhos não reflete necessariamente uma superioridade ou inferioridade da atual proposta, visto que cada trabalho foi conduzido em estudos de caso diferentes, com estudantes, recursos e propriedades de curso diferentes. Portanto, a análise comparativa apresentada em relação aos trabalhos presentes na literatura tem por objetivo apenas situar os atuais percentuais de precisão em relação ao estado da arte, pois não há, nesse caso, como constatar uma absoluta superioridade da proposta.

Além disso, destaca-se a importância das tarefas de pré-processamento para a modelagem de estudantes, visto que o balanceamento de classes e seleção de atributos impactaram significativamente nos resultados. Portanto, entende-se que a metodologia proposta obteve êxito, e pode ser replicada em outros trabalhos. Ademais, assume-se que os padrões de comportamento avaliados foram suficientes para a predição de EA neste curso, visto que os algoritmos avaliados foram capazes de classificar os estudantes com alto grau de acurácia. Em destaque, os padrões de comportamento relacionados ao fórum de discussões mostraram-se relevantes para a construção de um modelo de classificação, de modo que foram selecionados por diversas vezes na tarefa de seleção de atributos.

Por fim, para o estudo de caso, conclui-se que o classificador kNN alcançou os melhores resultados de classificação em geral. Esta técnica apresentou eficácia para a modelagem de estudantes em todos os cenários investigados, e portanto,

proporcionaria um nível de precisão viável caso fosse aplicada individualmente para a predição de EA dos estudantes deste curso. No entanto, uma combinação com outros algoritmos mostra-se mais eficaz em alguns cenários. Além disso, no que se refere às dimensões do modelo Felder-Silverman, os resultados demonstraram que a dimensão de percepção foi a mais suscetível aos métodos de classificação, atingindo mais de 85% de precisão em sua predição. Como cada dimensão corresponde a uma tarefa de classificação distinta, tais informações podem ser relevantes para trabalhos futuros, de modo que abrem margem para a investigação de quais características do curso levaram ao sucesso (ou fracasso) da predição em determinada dimensão. Em caso de fracasso, pode-se, por exemplo, investigar componentes que faltem ao curso estudado e que poderiam influenciar mais significativamente para a predição quanto a esta dimensão.

6 | CONCLUSÕES

No que se refere à experimentação, este trabalho trouxe esclarecimento com relação ao uso de técnicas de aprendizado de máquina na tarefa de caracterização de perfis de aprendizado. Especificamente, foram comparados quatro métodos de aprendizado de máquina voltados à tarefa de classificação: aprendizado Bayesiano, aprendizado baseado em instâncias, redes neurais artificiais, e árvores de decisão. Os resultados demonstraram que o aprendizado Bayesiano tem o melhor desempenho nas dimensões de processamento e de organização, ao passo que o aprendizado baseado em instâncias teve o melhor desempenho nas dimensões de captação e percepção. Em comparação com outros trabalhos, pôde-se notar que estes dois métodos, aprimorados com pré-processamento, superaram alguns dos trabalhos mais recentes, e mostram-se viáveis para aplicação em um sistema educacional adaptativo. Nota-se, ainda, que o uso de um único classificador não é suficiente para todas as dimensões, de modo que o uso combinado do aprendizado Bayesiano e do aprendizado baseado em instâncias provê uma classificação mais apurada do que o que se observou até agora na literatura.

De maneira geral, este trabalho contribui para o estado da arte com resultados de diversos classificadores aplicados ao mesmo problema de predição de EA, visto que esta trata-se de uma lacuna em trabalhos da área. Além disso, são expostos novos resultados que podem guiar estudos futuros, obtidos por meio de uma metodologia bem definida para extração de características em um ambiente virtual de ensino, pré-processamento de dados e coleta de diferentes medidas de avaliação dos resultados. A metodologia proposta também é entendida como uma contribuição ao estado da arte, de modo que pode ser reproduzida em trabalhos futuros. Como proposta para trabalhos futuros, espera-se avaliar a mesma abordagem a partir de outros modelos

de estilos de aprendizagem propostos na literatura. Entende-se que outros estudos apresentam potencial para bons resultados em uma classificação de estudantes. Além disso, espera-se experimentar uma classificação considerando novas características a respeito de cada aluno, como por exemplo, opiniões emitidas nos fóruns de discussões e dados coletados de redes sociais. Outra lacuna aberta é a observação de quais padrões de comportamento avaliados são os mais preditivos, e quais destes estão diretamente relacionados a determinado estilo de aprendizagem. Por fim, como complemento da pesquisa, abre-se a possibilidade de uma avaliação dos modelos de classificação gerados quanto à recomendação de conteúdo aos estudantes, ou seja, uma análise experimental do quanto a detecção automática de EA influencia o desempenho dos alunos durante o aprendizado em um sistema educacional personalizado.

REFERÊNCIAS

ABDULLAH, Manal Abdulaziz. **Learning style classification based on student's behavior in moodle learning management system**. Transactions on Machine Learning and Artificial Intelligence, v. 3, n. 1, p. 28, 2015.

CHA, Hyun Jin et al. **Learning styles diagnosis based on user interface behaviors for the customization of learning interfaces in an intelligent tutoring system**. In: International Conference on Intelligent Tutoring Systems. Springer, Berlin, Heidelberg, 2006. p. 513-524.

COFFIELD, Frank et al. **Learning styles and pedagogy in post-16 learning: A systematic and critical review**. 2004.

DORÇA, Fabiano Azevedo et al. **Detecção e correção automática de estilos de aprendizagem em sistemas adaptativos para educação**. Revista de Informática Teórica e Aplicada, v. 18, n. 2, p. 178-204, 2011.

DORÇA, Fabiano Azevedo et al. **Consistent evolution of student models by automatic detection of learning styles**. IEEE Latin America Transactions, v. 10, n. 5, p. 2150-2161, 2012.

DUNG, Pham Quang; FLOREA, Adina Magda. **An approach for detecting learning styles in learning management systems based on learners' behaviours**. In: International Conference on Education and Management Innovation. 2012. p. 171-177.

FELDER, Richard M. et al. **Learning and teaching styles in engineering education**. Engineering education, v. 78, n. 7, p. 674-681, 1988.

FELDMAN, Juan; MONTESERIN, Ariel; AMANDI, Analía. **Automatic detection of learning styles: state of the art**. Artificial Intelligence Review, v. 44, n. 2, p. 157-186, 2015.

GARCÍA, Patricio et al. **Evaluating Bayesian networks' precision for detecting students' learning styles**. Computers & Education, v. 49, n. 3, p. 794-808, 2007.

GRAF, Sabine. **Adaptivity in learning management systems focussing on learning styles**. 2007.

GRAF, Sabine et al. **Supporting teachers in identifying students' learning styles in learning management systems: An automatic student modelling approach**. Journal of Educational Technology & Society, v. 12, n. 4, p. 3, 2009.

JONASSEN, David H.; GRABOWSKI, Barbara L. **Handbook of individual differences, learning, and instruction**. Routledge, 2012.

KEEFE, J. W. **Learning style: An overview. In national association of their relationship.** British Journal of Educational Psychology, v. 67, p. 199-212, 1979.

KOLB, David A. **Experiential learning: Experience as the source of learning and development.** FT press, 1976.

LIYANAGE, M. Prabhani Pitigala; GUNAWARDENA, KS Lasith; HIRAKAWA, Masahito. **Using Learning Styles to Enhance Learning Management Systems.** ICTer, v. 7, n. 2, 2014.

MYERS, Isabel Briggs. **The Myers-Briggs Type Indicator: Manual.** 1985.

PASK, Gordon. **A fresh look at cognition and the individual.** International Journal of Man-Machine Studies, v. 4, n. 3, p. 211-216, 1972.

SENA, Edson Batista et al. **Uma abordagem computacional para detecção automática de estilos de aprendizagem utilizando modelos ocultos de markov.** SIED: EnPED-Simpósio Internacional de Educação a Distância e Encontro de Pesquisadores em Educação a Distância, 2016.

SOLOMAN, Barbara A.; FELDER, Richard M. **Index of learning styles questionnaire.** NC State University. Available online at: <http://www.engr.ncsu.edu/learningstyles/ilsweb.html> (last visited on 14.05.2010), v. 70, 2005.

TRUONG, Huong May. **Integrating learning styles and adaptive e-learning system: Current developments, problems and opportunities.** Computers in human behavior, v. 55, p. 1185-1193, 2016.

DETECÇÃO AUTOMÁTICA E DINÂMICA DE ESTILOS DE APRENDIZAGEM EM SISTEMAS ADAPTATIVOS E INTELIGENTES PARA A EDUCAÇÃO UTILIZANDO DYNAMIC SCRIPTING

Júlio César da Costa Silva

Universidade de Uberaba (UNIUBE)
Guanhães - Minas Gerais

Cristiano Grijó Pitanguí

Universidade Federal de São João del-Rei
(UFSJ)
São João Del Rei – Minas Gerais

Alessandro Vivas Andrade

Universidade Federal dos Vales do
Jequitinhonha e Mucurí (UFVJM)
Diamantina – Minas Gerais

Luciana Pereira de Assis

Universidade Federal dos Vales do
Jequitinhonha e Mucurí (UFVJM)
Diamantina – Minas Gerais

Cristiano Maciel da Silva

Universidade Federal de São João del-Rei
(UFSJ)
São João Del Rei – Minas Gerais

RESUMO Sistemas Adaptativos e Inteligentes para a Educação(SAIE) buscam fornecer assistência personalizada ao aluno por meio da detecção de seu Estilo de Aprendizagem (EA). Uma das técnicas mais exploradas para esta detecção é o Aprendizado por Reforço (AR). Contudo, tal técnica, em alguns casos, é considerada de lenta convergência. Este trabalho propõe o aperfeiçoamento de um SAIE utilizando uma adaptação da técnica de *Dynamic*

Scripting, considerada mais rápida que o AR. Experimentos demonstraram que a proposta, comparada às soluções da literatura, reduziu o número de problemas de aprendizagem em $\approx 54\%$, e $\approx 35\%$, respectivamente, para EAs Estáticos e Dinâmicos, enquanto reduz o número de interações do sistema em $\approx 5,7\%$.

PALAVRAS-CHAVE Dynamic Scripting. SAIE. Estilos de Aprendizagem.

ABSTRACT Adaptive and Intelligent Educational Systems (AIES) aim to provide a personalized assistance to the student by detecting their Learning Style (LS). One of the most popular techniques used for this detection is Reinforcement Learning (RL). However, RL presents slow convergence rate in some cases. This work proposes the improvement of an AIES by using an adaptation of the Dynamic Scripting technique, considered faster than the RL. Experimental results showed that the proposed technique reduced the number of the learning problems by $\approx 54\%$, and $\approx 35\%$, respectively, to Static and Dynamic LS, while reduced the number of interactions by $\approx 5.7\%$ in relation to the literature.

KEYWORDS Dynamic Scripting. Adaptive and Intelligent Educational Systems. Style.

1 | INTRODUÇÃO

A Educação à Distância (EAD) é um processo de ensino-aprendizagem mediado por Tecnologias da Informação e Comunicação (TICs) que se encontra em pleno desenvolvimento (KENSKI, 2003). As TICs viabilizaram os Ambientes Virtuais de Aprendizagem (AVA) que reúnem na *Web* diversas ferramentas, como fóruns, *wikis* e outras tecnologias colaborativas. Apesar da diversidade de ferramentas, elas não são adaptadas individualmente aos alunos. Assim, sistemas utilizando-se de técnicas de Inteligência Artificial (IA) são construídos para garantir tratamento adaptado a cada aluno (DORÇA, 2013).

Uma das formas de se gerar conteúdo adaptado aos alunos passa, primeiro, pela detecção dos Estilos de Aprendizagem (EA). A teoria dos EA presume que cada aluno tem características próprias que o distingue dos outros indivíduos (SILVA, 2012). Logo, cada aluno desenvolve uma maneira própria de processar as informações que recebe, sendo este processo denominado Estratégias de Aprendizagem (PRICE, 2004). Contudo, por mais pessoais que sejam essas estratégias, ainda é possível estabelecer padrões entre elas. Estes padrões são chamados de Estilos de Aprendizagem (FELDER, SILVERMAN, 1988; SILVA, 2012).

Os EAs permitem a criação de ambientes adaptativos, como o Sistema Adaptativo e Inteligente para Educação (SAIE) elaborado por Dorça (2012). Seu trabalho objetiva apresentar uma solução estocástica para provimento de adaptatividade e customização de Sistemas Educacionais por meio da modelagem probabilística dos EAs. Em síntese, seu sistema visa modelar o estudante, coletando e atualizando seus dados, de forma a descobrir seu EA. Com esta finalidade, o sistema, durante suas iterações, submete o aluno a avaliações. Notas das avaliações insatisfatórias, geram atualização do modelo do estudante por meio de Aprendizado por Reforço (AR) (DORÇA, 2013).

Apesar do AR ser uma técnica muito utilizada, Spronck (2005) afirma que ela pode ser considerada uma técnica “lenta” de aprendizado já que, em geral, demanda mais tempo para ajustar o elemento a ser otimizado, pois não há direção para o aprendizado. Esta lentidão pode prejudicar o sistema em convergir para o EA do aluno, denominado de Estilo de Aprendizado real (EAR). Delongas na convergência implicam na geração de Problemas de Aprendizagem (PAs), isto é, resultados insatisfatórios na avaliação. Significa uma maior dificuldade do aluno em aprender os conceitos ensinados. Assim, faz-se necessário o uso técnicas computacionais que aprimorem a convergência do sistema.

Spronck (2006) propõem o *Dynamic Scripting* (DS), uma técnica de Aprendizagem de Máquina *online*, isto é, aprendizado concomitante à própria utilização do sistema, que tem como características rapidez, eficácia, robustez, eficiência, clareza, variedade, consistência e escalabilidade. Spronck (2005) usou DS para tornar a IA de jogos adaptativa, obtendo notáveis resultados em seus experimentos.

O presente trabalho busca aprimorar o SAIE apresentado por Dorça, via substituição do componente de AR por uma adaptação do DS, visando aumentar a velocidade de convergência do sistema e reduzir os PAs frutos da demora dos ajustes realizados pelo AR. Resultados experimentais permitiram validar o uso da adaptação do DS em cenários de EAr Estáticos, em que o EAr não se modifica durante as iterações do sistema, e EAr Dinâmicos, em que o EAr é modificado durante as iterações do sistema. Em testes com EAr Estáticos, a média de redução dos PAs foi de 54.2% em relação à abordagem da literatura. Já em relação EAr Dinâmicos, obteve-se uma redução média de 35.8% dos PAs. Verificou-se, portanto, que a proposta alcança resultados promissores.

Este trabalho se organiza como segue. A seção 2 apresenta o referencial teórico e os principais trabalhos relacionados a esta pesquisa. A seção 3 apresenta de que forma o DS foi adaptado ao conceito de SAIE. Os resultados experimentais são apresentados na seção 4. Finalmente, a seção 5 conclui o trabalho e aponta alguns trabalhos futuros.

2 | REFERENCIAL TEÓRICO

2.1 Estilos de aprendizagem

Kolb (2005) define EA como a preferência de gerenciamento do aluno para trabalhar nas diferentes fases do ciclo da aprendizagem. Por sua vez, Felder e Silverman (1988) definiram EA como o método preferencial que as pessoas optam para receber e processar a informação.

A preocupação em se conhecer os EAs está em impedir que a divergência entre o EAr e a maneira como o conteúdo é exibido dificulte o aprendizado. Busca-se, portanto, uma sintonia entre o EAr e a forma como o conteúdo é exibido, com o objetivo de melhorar a produtividade do aluno no processo de ensino-aprendizagem.

Não é pacífico na literatura a validade da teoria dos EAs. Há trabalhos que enumeram possíveis falhas e controvérsias inerentes a esse assunto. Veenman (2003), por exemplo, criticam o modelo de detecção dos EAs baseado em questionários, alegando que o estudante não está apto a dizer verdadeiramente o que lhe é melhor. Rawson (2017) ressalta ainda a baixa correlação entre o que é respondido em um formulário e a verdade, evidenciando a baixa credibilidade dessas respostas.

Kirschner (2013) apontam que a muitos dos EAs são determinísticos, onde ao aluno não é atribuído um EA baseado em um conjunto de pontuações em diferentes dimensões, mas é classificado em um grupo específico. Essa clusterização em determinados EAs leva à compreensão de um EA fixo, o que pode comprometer a motivação para se aplicar e se adaptar às circunstâncias diferentes (PASHLER, 2008).

Para Kirschner (2017), uma preferência por um determinado EA é sinal de

fraqueza em relação aos outros, logo, deve ser eliminada e não respeitada. Hood (2017) criticam ainda a existência de mais de 70 modelos de detecção e delimitação de EA que não são coerentes entre si e que comprometem toda a validade da teoria.

Apesar das críticas, pesquisas recentes (DORÇA, 2012; FALCI, 2017; GONCALVES, 2016; RODRIGUES, 2016; SENA, 2016; SILVA, 2017) assinalam uma tendência na aplicação de ferramentas automáticas de detecção dos EAs que não se amoldam aos parâmetros criticados. A exemplo, tem-se a detecção do EA fundamentada em dados coletados por sistemas durante o processo de ensino-aprendizagem, sendo o questionário opcional, útil apenas para fornecer uma direção inicial

Dorça (2012) ressalta o aspecto dinâmico e probabilístico da seleção dos EAs baseando-se no *Felder-Silverman's Learning Styles Model* (FSLSM). O FSLSM possui um caráter probabilístico e separa os EAs em quatro dimensões: a Percepção (Sensorial/Intuitivo), a Entrada de Informações (Imagem/Verbal), Processamento de Informação (Ativo/Reflexivo) e Compreensão (Sequencial/Global) (FELDER, 1988). As dimensões dividem-se em 2 pólos que indicam o EA do estudante. O FSLSM analisa as dimensões no intervalo $[-11, 11]$ através da distância que se encontra de cada pólo (TRUONG, 2015). Assim, cada aluno tem uma maior probabilidade de possuir um EA, mas tem, em probabilidade menor, afinidade com todos os estilos (SILVA, 2012; DORÇA, 2013).

2.2 O SAIE de dorça (2012) e suas adaptações

SAIEs são softwares de gestão de aprendizagem que coletam dados de interação do aluno e, através de técnicas de IA, inferem suas características. A partir de então, o sistema passa a adaptar o ensino com base nas inferências realizadas (DORÇA, 2012).

Dorça (2012) propôs um SAIE composto pelos Modelo do Estudante (ME), o Módulo Pedagógico (MP), e o Componente de Modelagem do Estudante (CME).

O ME contém o Estilo de Aprendizagem Probabilístico (EAp), o Estado cognitivo (EC), e os Objetivos de Aprendizagem (OAs). O EAp é armazenado por valores reais no intervalo $[0, 1]$, sendo estes valores a probabilidade da preferência de um ou outro EA de cada dimensão do FSLSM. O EC, baseado na Taxonomia de Bloom (BLOOM, 1956), versa sobre o nível de conhecimento do aluno em cada conceito a ser aprendido. Existem 6 Níveis Cognitivos (NC), sendo: Conhecimento, Compreensão, Aplicação, Análise, Síntese e Avaliação. O EC indica qual o NC atual do aluno em dado conceito. Por sua vez, os OAs indicam quais NCs o aluno deve atingir. A condição de parada do algoritmo é a igualdade do ECs e OAs para todos os conceitos aprendidos (DORÇA, 2012).

O MP seleciona a estratégia pedagógica mais apropriada conforme o ME, com base em uma Combinação de Estilos de Aprendizagem (CEA) definida pelas 4 dimensões do FSLSM. A cada interação, o MP seleciona estocasticamente uma CEA.

Há 2 pólos para cada uma das 4 dimensões, assim, no total, 2^4 CEAs. Dessa forma, o aluno pode, com maior ou menor chance, se enquadrar em qualquer das 16 CEAs possíveis (DORÇA, 2012).

O CME realiza a atualização dos EAp no ME, usando AR. Ocorrendo um PA, usa-se a nota e a distância do EA (DEA), isto é, a distância entre os pólos de cada dimensão do FLSM, para calcular o reforço. Então, aplica-se o reforço no EAp, atualizando-se as probabilidades de seleção das CEAs para a próxima interação (DORÇA, 2012).

O algoritmo 1 destaca a utilização do AR considerando os componentes do SAIE. Seus principais passos são discutidos em seguida.

Algoritmo 1: AR aplicado à modelagem automática de EA (DORÇA, 2012)
1 Início
2 inicialize o ME;
3 Repita
4 Selecione um conceito C a ser apresentado ao estudante;
5 Selecione, através do MP, uma CEA a ;
6 Execute a , apresentando adequadamente o objetos de aprendizagem que ensinam C ao estudante;
7 Avalie, através do CME, a <i>performance</i> do aluno no conceito C ;
8 Atualize o EC no conceito C ;
9 Calcule o reforço R ;
10 Atualize o EAp do ME;
11 Faça de s o próximo estado, dado por s' ;
12 até s seja o estado final;
13 Fim

O passo 2 inicializa as variáveis contidas no ME no estado s , sendo s a descrição dos EAp armazenados pelo ME. O passo 4 seleciona o conceito a ser ensinado. O passo 5 realiza a seleção da CEA com base no EAp. O passo 6 apresenta ao aluno o conceito selecionado com base a CEA. O passo 7 submete o aluno a uma avaliação. No passo 8, o EC do aluno no conceito selecionado é atualizado. O passo 9, com base na nota, calcula o valor do reforço. O passo 10 atualiza o EAp com base no reforço calculado no passo 9. O passo 11 incrementa o estado s para o próximo estado s' . Repete-se o processo descrito até que s seja estado final.

Haider (2010) julga que fatores aleatórios podem influenciar a avaliação do aluno. Isto quer dizer que o ME pode conter o EAp bem próximo do EAr, mas, mesmo assim, a nota do aluno ser insatisfatória. Caso este fato ocorra, o AR trabalhará no sentido de distanciar o EAp do EAr. Há, portanto, nesse caso, um erro na aplicação do

reforço.

Gonçalves (2016) preocupado com a incorreta aplicação de reforço, propôs uma abordagem que usa a série histórica das notas obtidas para cada CEA. A proposta acrescenta a média das notas no ME, permitindo que o MP venha a se valer de mais uma variável para maximizar a chance de uma escolha correta da apresentação do conteúdo. Inova, ainda, na aplicação de reforço positivo caso a nota obtida seja 90% ou maior.

Falci (2017), de posse das notas anteriores, categorizou o cálculo do reforço não apenas com base na nota da interação, mas em todas os resultados já coletados, permitindo ainda, que sejam aplicados reforços positivos, de acordo com a categoria em que a nota se enquadrou. Assim, diminuem-se as chances de aplicação indevida de reforço.

Rodrigues (2016) substituiu a seleção de CEAs com uso de Cadeias de *Markov*, por *Lógica Fuzzy*. Inova na clusterização das notas em 5 conjuntos difusos em uma escala de Muito Ruim a Muito Boa. Usa o histórico armazenado das médias oriundas de cada conjunto difuso, obtendo resultados promissores em cenários com EAr Estáticos.

Apesar da obtenção de bons resultados, abordagens que usam histórico podem comprometer o aspecto probabilístico do sistema, além de se tornarem inapropriadas para cenários de EAr Dinâmicos, uma vez que, o histórico não irá contribuir caso o EAr mude durante as interações. Cerqueira (2000), Dorça (2012) e Silva (2012) ressaltam que os EAr sofrem alterações (graduais ou abruptas), influenciadas pelo conteúdo estudado, pelo fruto do processo de amadurecimento, e/ou por qualquer outra razão desconhecida.

Este trabalho propõe a substituição do AR por uma adaptação do DS para atualização das probabilidades das CEAs. Objetiva-se, portanto, aumentar a velocidade de convergência do EAp para o EAr, bem como reduzir os riscos de aplicação de reforço indevido. Tal proposta visa ainda apresentar-se como uma alternativa robusta, sendo compatível e eficiente tanto para cenários de EAr Estáticos quanto Dinâmicos.

2.3 O dynamic scripting

Dynamic Scripting (DS) é uma técnica de AR *online*, isto é, aprendizado concomitante à própria utilização do sistema, para IA de jogos (SPRONCK, 2005).

AR é uma técnica de Aprendizado de Máquina que mapeia estados à ações (SUTTON, 1998). As ações afetam o estado do ambiente e geram recompensas, que são encarregadas de medir o desempenho do agente em uma atividade. O objetivo do agente é maximizar os valores das recompensas recebidas (ARMSTRONG, 2006). No AR o aprendizado é por tentativa e erro. Assim, dado um estado do ambiente, o agente escolhe uma ação e a executa. Como resultado desta ação, o agente muda para outro estado (ou permanece no mesmo estado) e recebe uma gratificação. Repetindo este

ciclo, o agente aprende quais ações deve tomar para receber as maiores recompensas (SUTTON, 1998).

Em geral, técnicas de AR não são eficientes para a aprendizagem *online*, pois demandam grande quantidade de tentativas para convergência (DAWSON, 2002). Portanto, são aptas para problemas em que as tentativas ocorrem num curto intervalo de tempo. O DS, por outro lado, é capaz de aprender a partir de poucas tentativas por meio do uso de *scripts* tradicionais de IA, que limitam o tamanho do espaço de estados (SPRONCK, 2005).

Scripts de IA para jogos consistem em uma sequência de regras, onde cada regra consiste em duas partes: uma condicional, que identifica um ou mais estados do jogo; e uma de ação, que descreve a ação a ser tomada caso a condição seja satisfeita (SPRONCK, 2005). Contudo, estes *scripts* não apresentam características adaptativas. Seu funcionamento é baseado em uma série de regras, já conhecidas, que respondem aos estados do jogo. Já o DS adiciona a capacidade de explorar a representação do espaço de estados (que os *scripts* produzem) para uma aprendizagem rápida e eficiente, ao mesmo tempo em que confia no conteúdo baseado em *scripts* para garantir que todo o comportamento adaptativo seja plausível e eficaz (SPRONCK, 2005).

O mecanismo do DS pode ser definido em cinco etapas (THAWONMAS et al., 2006):

1. Uma base de dados, constituída por um conjunto de regras, é atribuída a um agente;
2. Regras são selecionadas da base, de acordo com os pesos, para a criação do *script*;
3. O agente de IA batalha contra o jogador usando o conteúdo de seu *script*;
4. O peso de cada regra no *script* é atualizado de acordo com o resultado da batalha;
5. Vá para 2.

O DS usa uma base de dados onde cada regra tem um peso que determina sua probabilidade de seleção para um *script*. Ao final de cada batalha, calcula-se um valor de *fitness*, que representa a eficácia do comportamento do agente. A *fitness*, mensura o desempenho do agente, e é usada para atualizar o peso das regras do *script* (SPRONCK, 2005).

A função de atualização de peso altera o peso das regras nos *scripts*, de acordo com o valor de *fitness* obtido. Valores altos de *fitness* aumentam os pesos, e baixos valores de *fitness* os diminuem. Assim, regras que fazem os agentes funcionarem bem serão associadas aos pesos maiores, o que significa que essas regras serão selecionadas com maiores probabilidades (SPRONCK, 2005). Dessa forma, os agentes controlados pelo DS se adaptarão e se sairão melhor contra um determinado

jogador.

3 I ABORDAGEM PROPOSTA

Este trabalho apresenta uma adaptação da técnica de DS, chamada de *Adapted Dynamic Scripting* (ADS), para o SAIE proposto por Dorça (2012).

O ADS é composto por um conjunto de regras constituídas por uma (ou mais) condição, uma ação e um peso. As condições são construídas usando-se as variáveis nota obtida pelo estudante na etapa de avaliação e/ou a Distância do Estilo de Aprendizagem (DEA), podendo ser as duas variáveis ou apenas uma delas, dependendo do propósito de cada regra. As ações, consistem na aplicação de reforço positivo ou negativo no EAp contido no ME, podendo ser feito através de um percentual sobre os valores existentes ou mesmo através de uma constante. Já o peso, por sua vez, determina a probabilidade de seleção de cada regra, sendo o valor que determina a qualidade da regra, quanto maior o peso, mais relevante é a regra. De forma geral, a estrutura das regras são bastante simples. Alguns exemplos são:

- **Se** (nota > 95) **então** aplique reforço no EAp de 5%;
- **Se** (nota < 10) e (DEA < 0.2) **então** inverta os pólos do EAp;
- **Se** (nota > 95) **então** aplique reforço no EAp de 0.00456;
- **Se** (nota < 60) e (DEA < 0.1) **então** aplique reforço de -5%.

Para o ADS foram criadas 40 regras que constituem sua base de regras, sendo que 20 dessas regras são necessárias para se compor um *script*. O mecanismo do ADS pode ser definido como:

1. Gere um *script* selecionando probabilisticamente 20 regras da base de regras;
2. Avalie o aluno e execute o *script*;
3. Atualize os pesos de todas as regras.

O passo 1 realiza o sorteio das regras que serão incorporadas ao *script*, considerando seus pesos. Todas as 40 regras são inicializadas com peso = 2,5, ou seja, com a probabilidade de 2,5% de serem sorteadas. O passo 2 verifica se as condições das regras no *script* são satisfeitas. Para cada regra que tem sua condição satisfeita, sua ação é executada. O passo 3 atualiza os pesos de todas as 40 regras de acordo com a nota obtida pelo aluno. Tais ajustes são realizados pela Equação 1:

$$f(n, i) = \frac{1}{1000} * \left(n / \frac{-Sc_i}{S_i} \right) \quad (1)$$

onde: n é a nota obtida pelo aluno; $i \in \{1, 2, \dots, 20\}$ é o nº da regra, Sc_i é o nº de vezes que a condição da regra i foi satisfeita; S_i é o nº de vezes que a regra i foi sorteada.

A aplicação do valor de *fitness* dado pela Equação 1 segue a seguinte lógica, representada pela Figura 1 e comentado na sequencia:

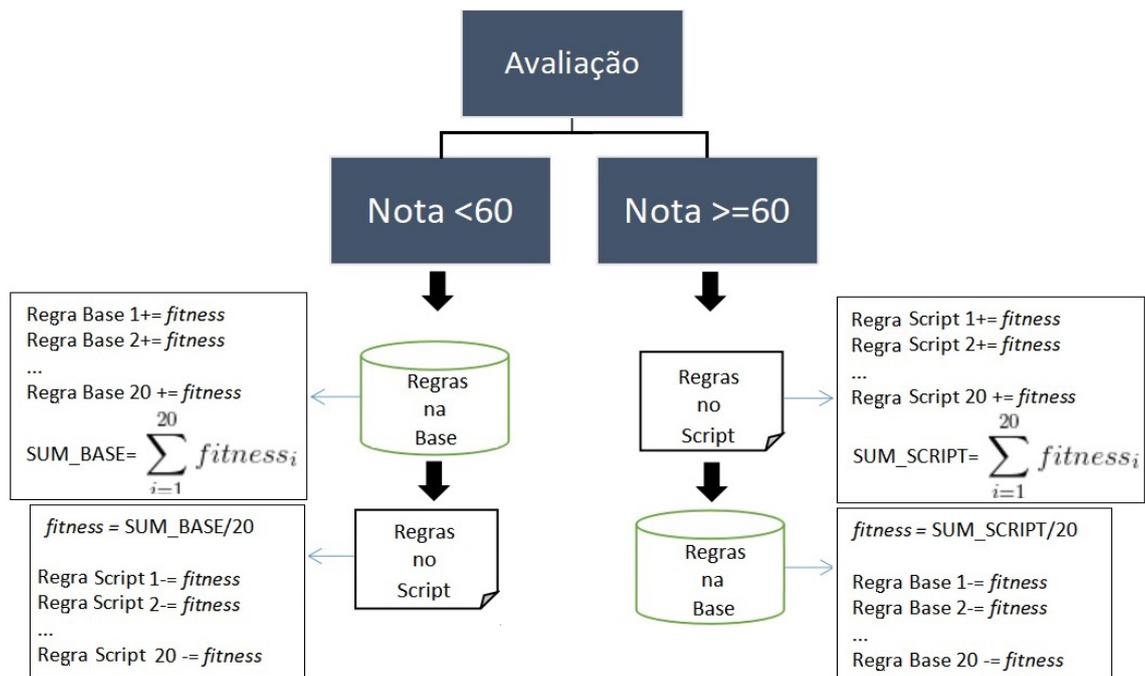


Figura 1- Demonstração da Atualização dos Pesos das Regras

Fonte: Elaborado pelo autor

- Caso a nota obtida pelo estudante seja ≥ 60 , os pesos das regras do script serão atualizados positivamente de acordo com seus dados (Sc_i e S_i) submetidos à equação 1, enquanto os pesos das regras da base de regras, que não foram selecionadas para o *script*, serão atualizados negativamente, reduzindo-se portanto, o valor dos seus pesos. Para isto, o somatório dos valores das atualizações positivas que foram realizadas em favor das regras que estavam no *script* é dividido por 20 (número de regras que não foram selecionadas para o *script*). O valor obtido nessa divisão será subtraído dos pesos das regras que não estão no *script*. Esta medida garante que a soma das probabilidades de seleção das 40 regras criadas seja sempre igual a 1.
- Caso a nota obtida pelo estudante seja < 60 , ocorre o contrário, sendo que os pesos das regras que ficaram na base de regras, isto é, que não foram selecionadas para o *script* serão atualizados positivamente de acordo com seus dados (Sc_i e S_i) submetidos à equação 1. Por outro lado os pesos das regras que estão no *script* serão atualizados negativamente de acordo com o somatório dos valores das atualizações positivas que foram realizadas em favor das regras que estavam na base de regras dividido por 20 (número de regras no *script*). O valor obtido nessa divisão será subtraído dos pesos

das regras que estão no script, penalizando-as pelo resultado insatisfatório.

Dessa forma, regras associadas aos maiores pesos, obtidos através de boas notas, serão consideradas mais apropriadas e, portanto, terão maiores chances de serem selecionadas para a criação do *script*.

O algoritmo 1, para acomodar a utilização do ADS, foi alterado em dois de seus pontos, a saber: antes do passo 4, acrescentou-se o passo 1 aqui apresentado. O passo 9 foi substituído pelos passos 2 e 3 aqui apresentados.

4 | SESSÃO EXPERIMENTAL

4.1 Metodologia experimental

O ADS se propõe a ser eficaz tanto para casos de EAr Estáticos quanto para EAr Dinâmicos. Assim, para validar a proposta, foram realizados testes comparativos com as abordagens de (DORÇA, 2012) e (FALCI, 2017) para EAr Estáticos e, com a abordagem de (DORÇA, 2012) para EAr Dinâmicos. Não foram comparadas as abordagens de Falci (2017), Gonçalves (2016), e Rodrigues (2016) para EAr Dinâmicos, pois, como falado na seção 2.2, tais abordagens não são apropriadas para estes casos.

Objetiva-se demonstrar que o ADS, contribui para se elevar o desempenho do estudante a partir da personalização do ensino, sendo que os indicadores utilizados para demonstrar essa contribuição são a redução na quantidade de PA encontrados durante o processo de aprendizagem e o aumento da média das notas dos estudantes.

Os parâmetros utilizados nos testes são os definidos por Dorça (2012), a saber: nota mínima para aprovação: 60%; quantidade de conceitos a serem aprendidos: 60; EAp inicial carregado no ME: Ativo, Sensitivo, Visual e Sequencial.

Os testes para EAr Dinâmicos foram realizados mudando o EAr, que é inserido no sistema no início da execução, a cada 150 interações. Assim, o sistema é iniciado com um EAr, e após 150 interações, esse EAr é alterado invertendo-se os pólos das 2 últimas dimensões. Em seguida, executam-se mais 150 interações e, alteram-se, novamente, os outros 2 pólos do EAr que não haviam sido alterados. Assim, inicia-se com um EAr e após 300 interações o sistema buscará convergir para um EAr oposto ao buscado inicialmente.

As informações referentes às notas foram obtidas calculando-se a média de todas as notas em cada teste, depois a média das médias dos 30 testes de cada experimento, realizando essa rotina para todas as 16 CEAs.

Realizaram-se 30 testes para cada CEA possível, sendo $2^4 * 30$ testes para cada abordagem com EAr Estáticos e EAr Dinâmicos.

4.2 Resultados experimentais

4.2.1 EAr Estáticos

A tabela 1 apresenta os resultados obtidos nos testes considerando EAr Estáticos. Cada experimento corresponde a uma CEA possível. As colunas que contém o símbolo %, correspondem à variação das outras abordagens em relação ao ADS.

	Dorça	Falci	ADS	Dorça	Falci	Dorça	Falci	ADS	Dorça	Falci
	Média Interações			%	%	Média PA			%	%
Exp. 1	383.8	361.2	366.2	-4.5	1.4	23.8	1.2	6.2	-73.6	423.3
Exp. 2	396	366.1	378.2	-4.4	3.3	36.0	6.1	18.2	-49.3	196.2
Exp. 3	394.8	368.3	371.3	-5.9	0.8	34.8	8.3	11.3	-67.3	37.1
Exp. 4	394.7	520.1	386.0	-2.2	-25.7	34.7	160.1	26.0	-25	-83.7
Exp. 5	395.2	365.7	376.4	-4.7	2.9	35.2	5.7	16.4	-53.2	185.2
Exp. 6	405.4	520.9	378.7	-6.5	-27.3	45.4	160.9	18.7	-58.8	-88.3
Exp. 7	402.5	515.1	379.2	-5.7	-26.3	42.5	155.1	19.2	-54.7	-87.6
Exp. 8	402.6	369.7	375.4	-6.7	1.5	42.6	9.7	15.4	-63.9	57.6
Exp. 9	405.5	364.5	380.3	-6.2	4.3	45.5	4.5	20.3	-55.3	352.2
Exp. 10	401.9	518.9	385.7	-4	-25.6	41.9	158.9	25.7	-38.5	-83.7
Exp. 11	403	521	377.7	-6.2	-27.5	43.0	161	17.7	-58.7	-88.9
Exp. 12	410.7	367.8	381.2	-7.1	3.6	50.7	7.8	21.2	-58.1	172.4
Exp. 13	408.9	524.5	386.4	-5.5	-26.3	48.9	164.5	26.4	-45.9	-83.9
Exp. 14	411.1	368	390	-5.1	5.9	51.1	8	30	-41.2	272.2
Exp. 15	413.3	368.2	380.9	-7.8	3.4	53.3	8.2	20.9	-60.7	152.9
Exp. 16	419.8	364.3	382.1	-8.9	4.8	59.8	4.3	22.1	-62.9	407.3
Média	403.1	424	379.7	-5.7	-7.9	43.1	64	19.7	-54.1	-69.1

Tabela 1. Comparativo para EAr Estático

Considerando os PAs, o ADS obteve uma redução de 54,1% em comparação a Dorça (2012) e de 69,1% em relação a Falci (2017). Destaca-se o Experimento 1 que apresentou uma redução nos PA de 73.6%, em relação a Dorça (2012). Nesse exemplo, o EAp inicial é igual ao EAr e, pode-se verificar o quanto o ADS contribuiu para acelerar a convergência, reduzindo os erros de reforço aplicados pelo AR.

Quanto ao número de interações, o ADS obteve uma redução de 5,7% comparado a Dorça (2012) e de 7,9% a Falci (2016). É importante notar que embora a redução do nº de iterações possa parecer pequena, em casos de aplicações reais, tais reduções são consideráveis. A exemplo, a UNIUBE, em seu modelo pedagógico de EAD de graduação, cada seção de aprendizagem (interação) possui uma semana de duração (UBERABA, 2016). Logo, essa redução, do ponto de vista real, significa um ganho considerável.

No que tange à nota média obtida nos experimentos com EAr Estáticos com

carga de 60 conceitos, a Figura 2 apresenta dados comparativos contendo os 16 Experimentos para cada abordagem. Não foi possível comparar as notas com as outras abordagens pois não realizaram esses testes.

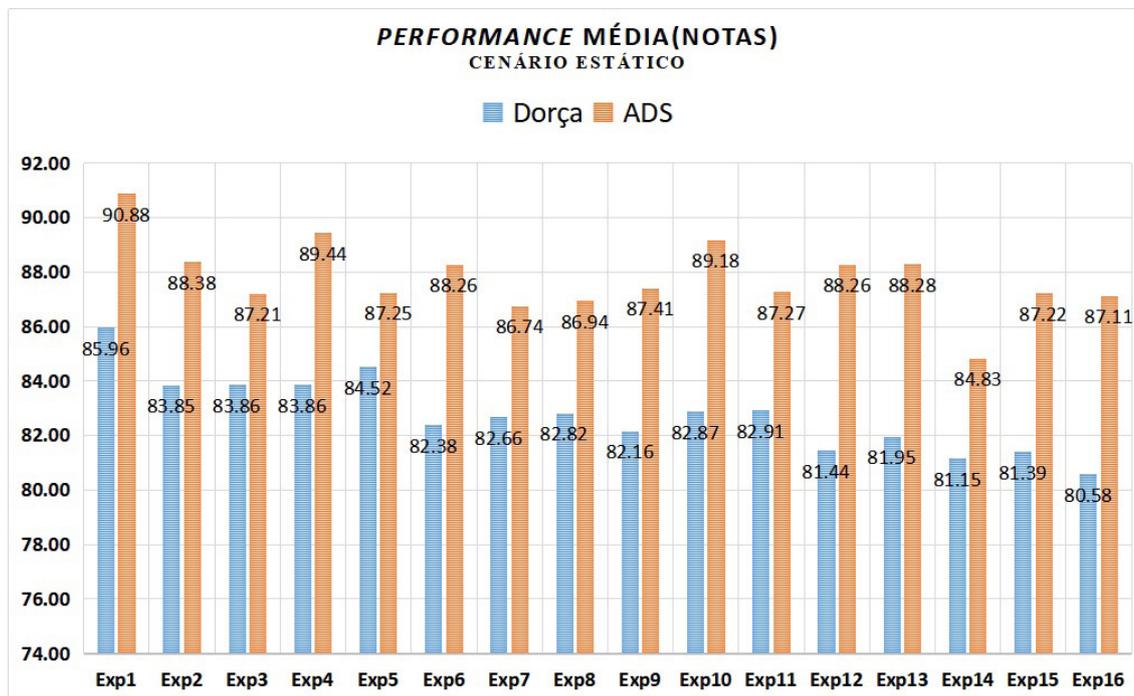


Figura 2: Comparação entre as abordagens no que tange à Média das Notas obtidas em cenários com EAr Estático

Fonte: Elaborado pelo autor

O ADS promoveu um aumento na nota média dos estudantes em 6.06% em comparação à abordagem da literatura, permitindo constatar que o ADS, além de reduzir os PAs se mostra apto a permitir um aumento na nota média dos estudantes

4.2.2 EAr Dinâmicos

A tabela 2 apresenta os resultados obtidos nos testes com EAr Dinâmicos, onde, considerando-se os PAs, o ADS apresenta uma redução de 35,8% em comparação a Dorça (2012). Destaca-se o Experimento 13 que apresentou uma redução de 48,9%, quase a metade dos PAs.

	Dorça	ADS	%	Dorça	ADS	%
	Média Interações			Média PA		
Exp. 1	418.2	393	-6	58.2	33	-43.2
Exp. 2	426.6	404	-5.2	66.6	44	-33.8
Exp. 3	428.7	412.1	-3.8	68.7	52.1	-24.2
Exp. 4	424.8	406.9	-4.2	64.8	46.9	-27.6
Exp. 5	423.5	397.4	-6.1	63.5	37.4	-41
Exp. 6	433.8	404	-6.8	73.8	44	-40.4
Exp. 7	433.5	417.2	-3.7	73.5	57.2	-22.1
Exp. 8	428.1	396.5	-7.3	68.1	36.5	-46.3

Exp. 9	431.6	422.2	-2.1	71.6	62.2	-13
Exp. 10	431.0	397.7	-7.7	71	37.7	-46.8
Exp. 11	430.3	402.5	-6.4	70.3	42.5	-39.5
Exp. 12	440.5	421.8	-4.2	80.5	61.8	-23.2
Exp. 13	434.0	397.7	-8.3	74	37.7	-48.9
Exp. 14	435.3	406.1	-6.6	75.3	46.1	-38.6
Exp. 15	437.7	402.1	-8.1	77.7	42.1	-45.8
Exp. 16	443.6	411.6	-7.2	83.6	51.6	-38.2
Média	431.3	405.8	-5.9	71.3	45.8	-35.8

Tabela 2. Comparativo para EAr Dinâmico

Em relação ao número de interações, o ADS apresentou uma redução de 5,9% para alcançar os OAs. É satisfatório o desempenho do ADS em relação à Dorça (2012), haja visto que, uma interação, em casos reais, demanda tempo considerável.

A nota média obtida nos experimentos com EAr Dinâmico, é apresentada na Figura 3 com um gráfico comparativo entre os 16 Experimentos de cada abordagem. Novamente o ADS elevou a nota média dos estudantes em comparação à Dorça (2012), mesmo em cenário adverso com oscilação do EAr do estudante o ADS se manteve capaz de permitir a maximização do desempenho do estudante.

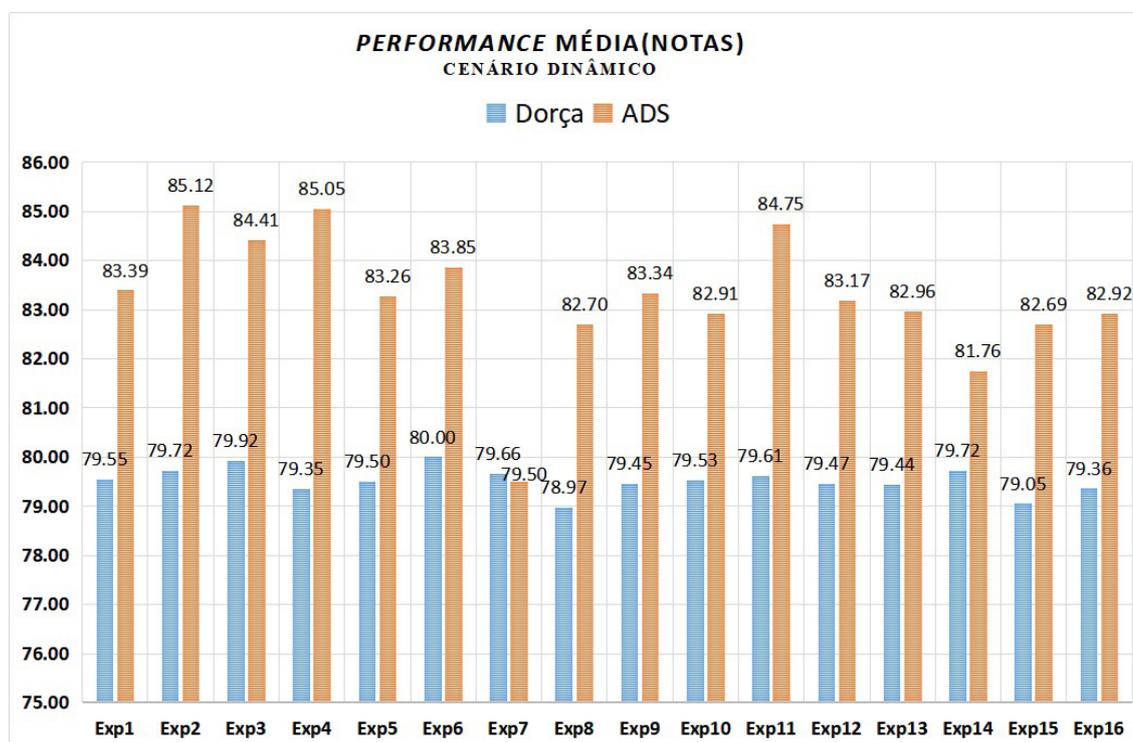


Figura 3: Comparação entre as abordagens no que tange à Média das Notas obtidas em cenários com EAr Dinâmico

Fonte: Elaborado pelo autor

5 | CONCLUSÃO E TRABALHOS FUTUROS

Este trabalho apresentou o ADS, uma adaptação da técnica de DS para aprimorar a convergência de um SAIE, com vistas a detectar e atualizar o EA do estudante para oferecer ao aluno um ensino personalizado. Levou-se em consideração que o EA do aluno muda ao longo do processo de ensino aprendizagem, necessitando de uma abordagem robusta que fosse apta a lidar com essa situação. Resultados experimentais apontaram redução na quantidade de interações bem como no número de PA em relação a trabalhos da literatura.

Como trabalho futuro pretende-se que o sistema aprenda automaticamente a estrutura das regras que são utilizadas pelo ADS, e não somente seus pesos.

REFERÊNCIAS

- ARMSTRONG, Warren et al. **Dynamic algorithm selection using reinforcement learning**. In: Integrating AI and Data Mining, 2006. AIDM'06. International Workshop on. IEEE, 2006. p. 18-25.
- BLOOM, B. S. et al. **Taxonomy os educational objectives**. New York: David mckay co. Inc. 1956.
- CERQUEIRA, Teresa Cristina Siqueira et al. **Estilos de aprendizagem em universitários**. 2000.
- DAWSON, Chad; FORMATIONS, S. Rabin. **AI Game Programming Wisdom**. Charles River Media. Inc., Hingham, Massachusetts, USA, p. 272-281, 2002.
- DE SENA, Edson Batista et al. **Uma abordagem computacional para detecção automática de estilos de aprendizagem utilizando modelos ocultos de markov**. SIED: EnPED-Simpósio Internacional de Educação a Distância e Encontro de Pesquisadores em Educação a Distância, 2016.
- DORÇA, Fabiano Azevedo et al. **Uma abordagem estocástica baseada em aprendizagem por reforço para modelagem automática e dinâmica de estilos de aprendizagem de estudantes em sistemas adaptativos e inteligentes para educação a distância**. 2012.
- DORÇA, Fabiano A. et al. **Comparing strategies for modeling students learning styles through reinforcement learning in adaptive and intelligent educational systems: An experimental analysis**. Expert Systems with Applications, v. 40, n. 6, p. 2092-2101, 2013.
- FALCI, Samuel Henrique et al. **Uma Nova Abordagem para Aplicação de Reforço em Sistemas Automáticos e Adaptativos de Detecção de Estilos de Aprendizagem**. Revista Eletrônica Argentina-Brasil de Tecnologias da Informação e da Comunicação, v. 1, n. 6, 2017.
- FELDER, Richard M. et al. **Learning and teaching styles in engineering education**. Engineering education, v. 78, n. 7, p. 674-681, 1988.
- GONÇALVES, André Vinícius. **Modelagem automática e dinâmica de estilos de aprendizagem em sistemas adaptativos e inteligentes para educação a distância: estudo comparativo entre duas abordagens**. 2016. Dissertação de Mestrado. UFVJM.
- HAIDER, M.; SINHA, A.; CHAUDHARY, B. **An Investigation of relationship between learning styles and performance of learners**. International Journal of Engineering Science and Technology, v. 2, n. 7, p. 2813-2819, 2010.
- HOOD, B. et al. **No evidence to back idea of learning styles**. The Guardian, March. 2017.

- KENSKI, Vani Moreira. **Tecnologias e ensino presencial e a distância**. Papirus Editora, 2003.
- KIRSCHNER, Paul A. **Stop propagating the learning styles myth**. *Computers & Education*, v. 106, p. 166-171, 2017.
- KIRSCHNER, Paul A.; VAN MERRIËNBOER, Jeroen JG. **Do learners really know best? Urban legends in education**. *Educational psychologist*, v. 48, n. 3, p. 169-183, 2013.
- KOLB, Alice Y. **The Kolb learning style inventory—version 3.1 2005 technical specifications**. Boston, MA: Hay Resource Direct, v. 200, p. 72, 2005.
- PASHLER, Harold et al. **Learning styles: Concepts and evidence**. *Psychological science in the public interest*, v. 9, n. 3, p. 105-119, 2008.
- PRICE, Linda. **Individual differences in learning: Cognitive control, cognitive style, and learning style**. *Educational Psychology*, v. 24, n. 5, p. 681-698, 2004.
- RAWSON, Kevin; STAHOVICH, Thomas F.; MAYER, Richard E. **Homework and achievement: Using smartpen technology to find the connection**. *Journal of Educational Psychology*, v. 109, n. 2, p. 208, 2017.
- RODRIGUES, Luiz Henrique Silva et al. **Análise comparativa de novas abordagens para modelagem automática e dinâmica de estilos de aprendizagem em sistemas adaptativos e inteligentes para educação**. In: Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE). 2016. p. 1076.
- SILVA, Júlio César da Costa et al. **Detecção Automática e Dinâmica de Estilos de Aprendizagem em Sistemas Adaptativos e Inteligentes utilizando Dynamic Scripting**. In: Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE). 2017. p. 1327.
- SILVA, Lisliê Lopes Vidal. **Estilos e estratégias de aprendizagem de estudantes universitários**. 2012. Tese de Doutorado. Universidade de São Paulo.
- SPRONCK, Pieter Hubert Marie. **Adaptive game AI**. Maastricht university, 2005.
- SPRONCK, Pieter et al. **Adaptive game AI with dynamic scripting**. *Machine Learning*, v. 63, n. 3, p. 217-248, 2006.
- SUTTON, Richard S.; BARTO, Andrew G. **Reinforcement learning: An introduction**. Cambridge: MIT press, 1998.
- THAWONMAS, Ruck; OSAKA, Syota. **A method for online adaptation of computer-game ai rulebase**. In: Proceedings of the 2006 ACM SIGCHI international conference on Advances in computer entertainment technology. ACM, 2006. p. 16.
- TRUONG, Huong May. **Integrating learning styles and adaptive e-learning system: Current developments, problems and opportunities**. *Computers in human behavior*, v. 55, p. 1185-1193, 2016.
- UBERABA, Universidade de. **Manual do Professor Tutor**. Uberaba, MG. 2016.
- VEENMAN, Marcel VJ; PRINS, Frans J.; VERHEIJ, Joke. **Learning styles: Self-reports versus thinking-aloud measures**. *British Journal of Educational Psychology*, v. 73, n. 3, p. 357-372, 2003.

UM PROCESSO DE VALIDAÇÃO DE VARIÁVEIS COMPORTAMENTAIS DE AUTORREGULAÇÃO DA APRENDIZAGEM EM PLATAFORMAS DE LMS

Rodrigo Lins Rodrigues

Universidade Federal Rural de Pernambuco,
Departamento de Educação
Recife-PE

João Carlos Sedraz Silva

Universidade Federal do Vale do São
Francisco, Colegiado de Engenharia Civil
Juazeiro-BA

Jorge Luis Cavalcanti Ramos

Universidade Federal do Vale do São
Francisco, Colegiado de Engenharia de
Computação
Juazeiro-BA

Fernando da Fonseca de Souza

Universidade Federal de Pernambuco, Centro
de Informática
Recife-PE

Alex Sandro Gomes

Universidade Federal de Pernambuco, Centro
de Informática
Recife-PE

RESUMO O presente estudo visa investigar a aplicabilidade do modelo de Regressão Múltipla para a validação de variáveis comportamentais de autorregulação da aprendizagem em contexto de EAD. Foi realizada uma pesquisa com 408 participantes de cursos na modalidade EAD com idade média de 30 anos. A coleta dos dados foi feita através de uma abordagem híbrida entre questionário e variáveis de arquivos de

log, com o objetivo de analisar se a afirmação dos alunos, respondida através do questionário, era explicada pelo seu comportamento dentro da plataforma de LMS. Como resultado, foi validado um conjunto de treze variáveis significativas referentes ao comportamento de autorregulação da aprendizagem dos alunos na plataforma Moodle.

PALAVRAS-CHAVE Autorregulação da Aprendizagem, Regressão Múltipla, LMS.

1 | INTRODUÇÃO

Estudos vêm sendo desenvolvidos com o intuito de analisar a importância de habilidades de autorregulação para o aprendizado em ambientes de EAD (BARNARD et al., 2008); (ALI et al. 2014). A autorregulação da aprendizagem é a habilidade cognitiva por meio da qual as pessoas conseguem gerir recursos, o tempo, os esforços, o ambiente, a interação e a busca por ajuda de no processo de aprendizagem (CHEN, 2002); (PINTRICH, 1999); (ZIMMERMAN & MARTINEZ-PONS, 1988).

Estudantes que detêm destas habilidades conseguem controlar o ritmo, o direcionamento do processo de aprendizagem e gerenciar um conjunto de características do ambiente

de aprendizagem, realizando escolhas que o auxiliam no desenvolvimento de suas funções cognitivas (BROADBENT & POON, 2015).

Pesquisadores acreditam que os ambientes de aprendizagem on-line exigem do aluno uma maior consciência de seu processo de aprendizagem autorregulado (DABBAGH & KITSANTAS, 2005); (SCHUNK & ZIMMERMAN, 1998). Estes autores acreditam que as habilidades de autorregulação são essenciais para o sucesso da aprendizagem autônomas (autodirigidas) e que o desenvolvimento dessas habilidades pode ser suportado por ferramentas pedagógicas baseadas na Web (AZEVEDO, 2005); (DABBAGH & KITSANTAS, 2004).

No entanto, ainda existem poucas pesquisas que objetivam a mensuração de características comportamentais de autorregulação em ambientes virtuais de aprendizagem e cursos na modalidade de EAD. A maioria dos instrumentos de coleta utilizados na literatura foram concebidos para serem aplicados no ensino presencial.

Diante deste contexto este artigo tem como objetivo a identificação e validação de variáveis, em ambientes de LMS, que possibilitem identificar traços comportamentais de autorregulação da aprendizagem de alunos em cursos na modalidade de EAD, possibilitando assim que se possa aplicar técnicas de estatística e mineração de dados para tomadas de decisões em tempo real.

2 | ABORDAGENS DE MENSURAÇÃO DE SRL NA LITERATURA

Quantificar habilidades de autorregulação do aluno, durante as tarefas específicas de aprendizagem é um desafio. A maioria das abordagens de mensuração são realizadas através de coleta com questionários.

2.1 Questionários como instrumentos de coleta

Um dos instrumentos utilizados na literatura é o questionário MSLQ (YUKSELTURK & BULUT, 2009), (ADESOPE, ZHOU, & NESBIT, 2015), (PANADERO, KIRSCHNER, JÄRVELÄ, MALMBERG, & JÄRVENOJA, 2015), (SANCHEZ-SANTILLAN, PAULE-RUIZ, CERESO, & ALVAREZ-GARCÍA, 2016), (ALI, HATALA, GAŠEVIĆ, & WINNE, 2014). Este instrumento é um dos mais utilizado na literatura, embora uma de suas limitações seja o fato do mesmo ter sido construído para ser utilizado em ensino presencial, não levando em considerações aspectos peculiares da modalidade de EAD.

Outro instrumento utilizado na literatura foi desenvolvido por (BARNARD, LAN, TO, PATON, & LAI, 2009) intitulado OSLQ. Este instrumento foi desenvolvido e projetado para a avaliação de habilidades de autorregulação que são importantes na aprendizagem on-line. Suas subescalas incluem estruturação ambiente, estabelecimento de metas, gestão do tempo, a procura de atendimento, estratégias

de tarefas de auto-avaliação.

Embora os instrumentos de coleta através de questionários, sejam os mais utilizados pela literatura, existem várias limitações no que diz respeito a: necessita de vários dias para a coleta de dados, são baseadas na opinião do aluno sobre seu próprio comportamento e não levam em consideração as reais ações comportamentais do aluno no seu processo de ensino e aprendizagem.

Alguns autores questionam a validade das abordagens de mensuração através de questionários, estas abordagens caracterizam-se pela autoafirmação retrospectiva do aluno sobre suas ações de aprendizagem. Autores como (WINNE & PERRY, 2000) E (WINNE & JAMIESON-NOEL, 2002), tem duvidado da eficácia destas abordagens, pois afirmam que em alguns casos os alunos podem não serem capazes de lembrar com precisão suas atividades no passado, podem subestimar seu próprio comportamento dando respostas tendenciosas e em casos mais extremos podem não responder tais perguntas com honestidade.

2.2 Arquivos de log como instrumentos de coleta

Uma segunda abordagem de coleta de dados é através de arquivos de log. Os principais trabalhos que utilizam essa abordagem são os trabalhos de (SCHOOR & BANNERT, 2012), (SABOURIN, MOTT, & LESTER, 2012), (BONDAREVA, ET AL., 2013), (NUSSBAUMER, HILLEMANN, GÜTL, & ALBERT, 2015), (SEGEDY, KINNEBREW, & BISWAS, 2015), (SONNENBERG & BANNERT, 2015), (YOU, 2016). A maioria destes trabalhos utilizam dados de softwares educacionais específicos para o e-Learning e não focam em dados de plataformas de LMS.

Dentro da perspectiva de mensuração através de arquivos de *log* em plataformas de LMS, um dos trabalhos que merece destaque é o trabalho do (YOU, 2016). Neste estudo o autor identifica indicadores comportamentais para a aprendizagem autorregulada em sistemas de LMS tentando prever o sucesso dos alunos ao término de um curso. Neste trabalho o autor utilizou regressão linear para construção de modelos e verificação de variáveis significativas. Uma das limitações deste trabalho é a pouca profundidade teórica na escolha das variáveis explicativas.

Diante desta problemática, as próximas seções abordam o método desta pesquisa que busca identificar e validar as principais variáveis comportamentais em ambientes de LMS no que diz respeito aos construtos de autorregulação da aprendizagem. Estas variáveis possibilitarão que pesquisadores que tenham interesse neste fenômeno, possam realizar inferências em tempo real utilizando um conjunto de variáveis representativas do fenômeno.

3 | MÉTODO

Neste trabalho foi utilizado a técnica estatística de Análise de Regressão Linear Múltipla com dados de questionário e dados comportamentais extraídos de arquivos log.

Inicialmente foi aplicado o questionário de autorregulação da aprendizagem OSLQ (BARNARD, LAN, TO, & PATON, 2009). Em seguida foram identificadas e extraídas as variáveis comportamentais do banco de dados da plataforma Moodle através de logs, que representam os comportamentos dos alunos de acordo com os construtos definidos pela teoria de autorregulação da aprendizagem.

De posse de ambas as bases de dados, foi realizada uma integração, através da chave primária de ambas as bases de dados. A função desta fase de integração foi construir uma só base que contivesse dados do questionário e dados comportamentais, com o objetivo de analisar se a afirmação dos alunos, respondida através do questionário, era explicada pelo seu comportamento dentro da plataforma de LMS.

3.1 Público alvo e Amostra

Para a aplicação do instrumento, foi utilizada uma amostra coletada através da Universidade de Pernambuco que promove, a mais de dez anos, cursos de graduação na modalidade de EAD através do programa Universidade Aberta do Brasil.

Os dados foram coletados através de um processo de amostragem aleatória simples (BOLFARINE & BUSSAB, 2005), totalizando uma amostra de 408 alunos, com idade média de 30 (SD=18,23) anos coletada durante um período de dois meses. A Tabela 1 mostra a frequência do quantitativo de alunos por curso.

Cursos	Frequência	Freq. Rel.	Freq. Perc.	Freq. Acum.
Administração Pública	27	0,07	6,62	6,62
Licenciatura em Biologia	112	0,27	27,45	34,07
Licenciatura em Letras	88	0,22	21,57	55,64
Licenciatura em Pedagogia	170	0,42	41,67	97,30
Não Informado	11	0,00	2,70	100,00
Total	408			

Tabela 1. Variáveis de interação coletadas.

Dos alunos que fizeram parte da amostra 57,7% possuem apenas ensino médio, 24,22% já cursaram algum curso superior, seja completo ou parcialmente e 7,35% possuem pós-graduação, nos níveis especialização, mestrado ou doutorado.

3.2 Instrumentos de coleta Questionário OSLQ

O instrumento de coleta utilizado nesta pesquisa foi adaptado do questionário de autorregulação da aprendizagem on-line (OSLQ) desenvolvido por (BARNARD, LAN, CROOKS, & PATON, V. O., 2008) E (BARNARD, LAN, TO, PATON, & LAI, 2009). Este instrumento é uma escala de 24 itens com um formato de resposta do tipo escala de Likert de 5 pontos com valores variando de 'concordo totalmente' (5) até 'discordo totalmente' (1).

Como as competências de autorregulação da aprendizagem parecem ser “altamente dependentes de contexto” (SCHUNK, 2005), o desenvolvimento de um instrumento adaptado para os ambientes de aprendizagem on-line é uma etapa necessária em pesquisas que avaliam o fenômeno na modalidade de ensino à distância. Nas pesquisas de (BARNARD et al., 2009) a autora avaliou a manifestação das competências de autorregular a aprendizagem em ambientes de aprendizagem online com uma amostra de 434 estudantes de Universidades localizadas no sudoeste dos Estados Unidos. Para este estudo realizamos, após a aplicação do questionário, uma avaliação do mesmo, através da análise de adequação do instrumento utilizando os valores de *Alpha de Cronbach*.

3.3 Variáveis comportamentais de SRL

O questionário, descrito na seção anterior, serviu para que pudéssemos fazer um mapeamento de variáveis comportamentais na plataforma LMS Moodle, afim de obter variáveis que representassem os itens do questionário, não apenas de forma opinativa, mas como uma comprovação da opinião do aluno, através da extração do seu próprio comportamento nos registros de *log* da plataforma Moodle. Para tal, as variáveis foram mapeadas e agrupadas dentro dos construtos de autorregulação da aprendizagem, como mostrado na Tabela 1.

Variável	Descrição sobre as variáveis	Construto
VAR01	Quantidade de diferentes locais (IP's) a partir dos quais a(o) aluna(o) acessou o ambiente.	Estrut. do ambiente
VAR02	Quantidade de mensagens enviadas por aluna(o) às(os) Professoras(es) pelo ambiente.	Busca por Ajuda
VAR03	Quantidade de mensagens enviadas por aluna(o) às(os) Tutor(es) pelo ambiente.	
VAR04	Quantidade geral de mensagens enviadas pela(o) aluna(o) dentro do ambiente.	
VAR05	Quantidade geral de mensagens recebidas pela(o) aluna(o) dentro do ambiente.	
VAR06	Quantidade de tópicos criados pelo(a) aluno(a) em fórum do tipo "tira-dúvidas".	
VAR07	Quantidade de postagens no "Fórum tira dúvidas";	
VAR08	Quant. de postagens de um(a) aluno(a) em fóruns que foram respondidas por outros(as) alunos(as).	
VAR09	Quantidade de postagens de um(a) aluno(a) em fóruns que foram respondidas pelo(a) professor(a) ou tutor(a).	
VAR10	Quantidade de colegas diferentes para quem o(a) aluno(a) enviou mensagens dentro do ambiente.	

VAR12	Quantidade de visualizações da aba "Conteúdo" do curso, onde constam os arquivos com o conteúdo programático do curso	Estratégias para Realização de Tarefas
VAR13	Horário que mais realizou atividades;	
VAR14	Turno do dia em que realizou mais atividades.	
VAR15	Quantidade de atividades entregues por um(a) aluno(a) no prazo, por disciplina;	
VAR16	Quantidade de atividades entregues por um(a) aluno(a) fora do prazo, por disciplina;	
VAR17	Tempo médio entre a abertura da atividade e sua submissão;	
VAR18	Quantidade de leituras feitas ao fórum (pageviews);	
VAR19	Quantidade de postagem feitas ao fórum;	
VAR20	Quantidade de respostas ao tópico principal (refazer opinião em fórum);	Autoavaliação
VAR21	Quantidade de pageviews ao quadro de notas;	
VAR22	Quantidade de vezes que o aluno visualiza o (Cheklist Atividades)	
VAR23	Quantidade de visualizações de notas por atividade;	
VAR24	Média semanal da quantidade de acessos de um(a) aluno(a) ao ambiente.	Gerenciamento do tempo
VAR25	Tempo médio entre a criação de um tópico no fórum temático e a primeira postagem do aluno;	
VAR28	Quantidades de Time Out;	
VAR29	Tempo médio de realização de uma atividade;	
VAR30	Tempo médio entre a abertura da atividade e sua realização;	
VAR31	Quantidade de acessos do(a) aluno(a) ao ambiente.	
VAR31b	Quantidade de dias distintos que o aluno entrou na disciplina	
VAR31c	Quantidade de dias distintos que o aluno entrou na plataforma	
VAR32a	Quantidade de acessos do(a) aluno(a) ao ambiente por turno (Manhã).	
VAR32b	Quantidade de acessos do(a) aluno(a) ao ambiente por turno (Tarde).	
VAR32c	Quantidade de acessos do(a) aluno(a) ao ambiente por turno (Noite).	
VAR32d	Quantidade de acessos do(a) aluno(a) ao ambiente por turno (Madrugada).	
VAR33	Quantidade de atividades entregues por um(a) aluno(a) no prazo, por disciplina;	Estabelecimento de metas
VAR34	Quantidade geral de postagens de um(a) aluno(a) em fóruns.	
VAR35	Quantidade de respostas de um(a) professor(a) para as dúvidas de alunos(as) em fóruns.	

Tabela 1. Variáveis mapeadas do LMS Moodle.

Foram elaborados trinta e cinco *scripts SQL* para a extração das variáveis na base de dados do Moodle. Todas as variáveis comportamentais extraídas, foram referentes a comportamentos dos alunos durante o período de um ano. Em seguida foi realizada a junção dos dados coletados com o questionário e os dados comportamentais coletados através da base de dados do Moodle.

4 | ANÁLISE E INTERPRETAÇÃO DOS DADOS

Para as análises deste artigo foi utilizado o software estatístico R. Para a validação do questionário, através cálculo dos valores de *alpha de Cronbach*, foi utilizado o pacote *psy* onde posteriormente foi realizado a análise de Regressão Linear Múltipla.

4.1 Validação do questionário

O questionário é composto por 24 itens, onde é subdividido um quantitativo de itens que representa cada um dos seis construtos. Para a validação do questionário

foi utilizado os valores de Alpha de *Cronbach* com o intuito de verificar o índice de consistência interna dos construtos do questionário. Este índice foi inicialmente apresentado por Lee J. Cronbach (1951), como uma forma de verificar a confiabilidade de um instrumento de coleta tipo questionário.

Os resultados para os valores de alpha de *Cronbach* para cada construto foram: 'estabelecimento de metas' ($\alpha=0,59$), 'estruturação do ambiente' ($\alpha=0,79$), 'estratégia para realização de tarefas' ($\alpha=0,74$), 'gerenciamento do tempo' ($\alpha=0,75$), 'busca por ajuda' ($\alpha=0,74$) e 'autoavaliação' ($\alpha=0,72$).

É importante mencionar que não existe um consenso na literatura acerca da interpretação da confiabilidade de um questionário obtida a partir do valor deste coeficiente. Não há um valor mínimo definido para o coeficiente alfa de Cronbach ser aceito como bom, mas se acha na literatura o valor de 0,70 como aceitável (FREITAS & RODRIGUES, 2005); (URDAN, 2001); (OVIEDO & CAMPO-ARIAS, 2005); (MILAN & TREZ, 2005). Embora existam autores que utilizam o valor de alfa sem fazerem nenhuma menção a um mínimo aceitável (SALOMI, MIGUEL, & ABACKERLI, 2005); (MIGUEL & SALOMI, 2004); (MATO & VEIGA, 2000).

Nesta validação apenas o construto Estabelecimento de metas obteve valor de coeficiente *Alpha de Cronbach* abaixo de 0,07. Estes resultados foram importantes para darmos prosseguimento as análises com a aplicação da análise de regressão múltipla.

4.2 Validação das variáveis comportamentais

Para a validação das variáveis comportamentais foi aplicado o modelo de Regressão Múltipla, onde existe o interesse em avaliar a relação de uma variável Y (variável dependente ou variável resposta) em relação a k variáveis X_j (variáveis independentes ou covariáveis), $j = 1, 2, \dots, k$. O objetivo foi reduzir um grande número de variáveis para poucas dimensões com o mínimo de perda de informação, permitindo a detecção dos principais padrões de similaridade, associação e correlação entre as variáveis de acordo com os construtos (CAMERON, & TRIVEDI, 1998). Tal modelo é apresentado na seguinte equação:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_k x_k + \mathcal{E}_i, \quad i = 1, \dots, n \quad (1)$$

Em que, n é o número de indivíduos, y_i é a observação da variável dependente para o i -ésimo indivíduo, $x_i = (x_{i,1}, x_{i,2}, \dots, x_{i,k})'$ é um vetor de observações das variáveis independentes para o i -ésimo indivíduo, $\beta = (\beta_0, \beta_1, \beta_2, \dots, \beta_k)'$ é um vetor de coeficientes de regressão (parâmetros) e \mathcal{E}_i é um componente de erro aleatório. Assume-se que esses erros são independentes e seguem distribuição normal com média zero e variância desconhecida σ^2 .

Neste artigo a variável resposta (Y_i) é tida como a média dos itens do questionário que representa cada construto de autorregulação da aprendizagem e as

variáveis independentes X_n são as variáveis que representam os comportamentos de autorregulação dos alunos, extraídos da base de dados da plataforma Moodle.

Para a análise foi realizado a construção de seis modelos desenvolvidos referentes aos seguintes construtos: (1) Estabelecimento de Metas, (2) Estruturação do Ambiente (EA), (3) Estratégias para realização de Tarefas (ET), (4) Gerenciamento do Tempo (GT), (5) Busca Por Ajuda (BA) e (6) Auto Avaliação (AA).

Através do modelo de *Regressão Múltipla*, para cada um dos construtos, foi obtido o valor de significância para cada uma das variáveis. O objetivo foi identificar quais variáveis comportamentais melhor explicam o construto identificado através do questionário, ou seja, identificar os principais comportamentos de autorregulação dos alunos que explicassem suas afirmações através do questionário OSQL (BARNARD et al., 2009).

A tabela 2 mostra a variáveis mais significativas, que melhor explicam o construto.

Construto	Variável	Significância (<i>p</i> -valor)
Estruturação do Ambiente (EA)	VAR01	0,0366
Busca Por Ajuda (BA)	VAR02, VAR03, VAR04	0,0304; 0,0309; 0,0433;
Estratégias para realização de Tarefas	VAR18	0,0438
Auto Avaliação (AA)	VAR22	0,000435
Gerenciamento do Tempo (GT)	VAR28, VAR29,	0,044961; 0,000612; 0,004469;
Estabelecimento de Metas (EM)	VAR33, VAR34	4,91e-07; 0,0013

Tabela 2. Significância das variáveis para cada um dos construtos

*Variáveis com *p*-valor acima de 0,05 não possui significância estatística

O primeiro modelo especificado foi o modelo de *Estruturação do ambiente (EA)*. Este construto, apesar de existir no questionário, foi difícil apontar uma variável comportamental dentro do ambiente que pudesse representar a estruturação do seu próprio ambiente de trabalho do aluno. Neste caso apenas a variável (*VAR01*) foi extraída, que trata da quantidade de diferentes locais (*IP's*) a partir dos quais a(o) aluna(o) acessou o ambiente. Esta variável nos demonstra o quanto o aluno, durante um semestre, permanece estudando em um só local de trabalho ou em locais e dispositivos diversificados. O resultado deste modelo apontou uma significância estatística para esta variável (*p*-valor 0,0366).

O segundo modelo foi o que obteve o maior número de variáveis significativas, este modelo teve como variável dependente o construto *Busca por Ajuda (BA)*. Dentre todas as variáveis incluídas no modelo, cinco delas tiveram significância estatística (*VAR02, VAR03, VAR04, VAR05, VAR10*). Todas as variáveis significativas se referem a troca de mensagens privadas entre aluno e professor, aluno e tutor em busca de ajuda.

O terceiro modelo desenvolvido foi referente as *Estratégias para Realização de Tarefas (ET)*. Este modelo foi construído por oito variáveis comportamentais, que

inicialmente representavam o comportamento do aluno em traços dentro da plataforma que indicavam estratégias para realização de tarefas. Após a aplicação do modelo de *Regressão de Múltipla*, foi verificado que apenas a variável (*VAR18*) que trata da quantidade de leituras feitas ao fórum (pageviews) teve significância estatística para representar o construto de estratégias para realização de tarefas.

O quarto modelo foi relacionado ao construto *Auto Avaliação (AA)*. Inicialmente para este construto foram apontadas quatro variáveis comportamentais, no entanto apenas a variável (*VAR22*), que trata da quantidade de vezes que o aluno visualiza o checklist de atividades realizadas, essa funcionalidade permite o aluno consultar seu status de realização de atividades dentro da plataforma.

O quinto construto, referente ao *Gerenciamento do Tempo (GT)* foi o segundo modelo que teve mais variáveis significativas (*VAR28, VAR29, VAR31b, VAR31c*). A variável 28 e 29 são relacionadas ao tempo que o aluno gasta para fazer uma atividade e as variáveis 31b e 31c referem-se à quantidade de acessos do aluno ao ambiente.

O último modelo refere-se ao construto *Estabelecimento de Metas (EM)*. Para este construto, das três variáveis comportamentais escolhidas, duas delas (*VAR33* e *VAR44*) tem significância estatística para representar o construto estabelecimento de metas. A variável (*VAR33*) refere-se a quantidade de atividades entregues por um(a) aluno(a) no prazo. A variável 34 refere-se a Quantidade geral de postagens de um(a) aluno(a) em fóruns. De acordo com a análise, podemos afirmar que ambas as atividades têm significância estatística para representar o construto Estabelecimento de metas.

Ao final da aplicação dos seis modelos de regressão obtivemos um total de treze variáveis significativas referente ao comportamento de autorregulação da aprendizagem dos alunos na plataforma Moodle.

5 | CONSIDERAÇÕES FINAIS

Ainda é pouco expressiva a quantidade de pesquisas sobre a mensuração das características de autorregulação da aprendizagem com foco em ambientes de EAD. A maioria dos instrumentos de coleta disponibilizados na literatura para a mensuração de SRL foram desenvolvidos para ambientes presenciais de ensino.

Os resultados desta pesquisa vêm a contribuir com a comunidade acadêmica de Mineração de Dados Educacionais bem como a comunidade de pesquisadores que buscam entender os fenômenos de autorregulação da aprendizagem, especificamente em contextos da aprendizagem à distância.

A validação das variáveis comportamentais, possibilitará que outros pesquisadores possam aplicar métodos da estatística e mineração de dados para a construção de modelos preditivos capazes de dar indícios de previsibilidade ao professor para tomada de decisão e condução da ação docente, sobre o comportamento de

autorregulação da aprendizagem dos alunos.

REFERÊNCIAS

- Adesope, O. O., Zhou, M., & Nesbit, J. C. (2015). Achievement Goal Orientations and Self-Reported Study Strategies as Predictors of Online Studying Activities. *Journal of Educational Computing Research*, 436-458.
- Ali, L., Hatala, M., Gašević, D., & Winne, P. H. (2014). Leveraging MSLQ Data for Predicting Students Achievement Goal Orientations. *Journal of Learning Analytics*, 157-160.
- Azevedo, R. (2005). Using hypermedia as a metacognitive tool for enhancing student learning? The role of self-regulated learning. *Educational Psychologist*, 199-209.
- Barnard, L., Lan, W. Y., Crooks, S. M., & Paton, V. O., V. O. (2008). The relationship of epistemological beliefs with self-regulatory skills in the online course environment. *Journal of Online and Learning Teaching*.
- Barnard, L., Lan, W. Y., To, Y. M., & Paton, V. O. (2009). Measuring self-regulation in online and blended learning environments. *The Internet and Higher Education*, 1-6.
- Barnard, L., Lan, W. Y., To, Y. M., Paton, V. O., & Lai, S. L. (2009). Measuring self-regulation in online and blended learning environments. *The Internet and Higher Education*.
- Bolfarine, H., & Bussab, O. H. (2005). *Elementos de amostragem*. São Paulo: Edgard Blücher.
- Bondareva, D., Conati, C., Feyzi-Behnagh, R., Harley, J. M., Azevedo, R., & Bouchet, F. (2013). Inferring learning from gaze data during interaction with an environment to support self-regulated learning. In *Artificial Intelligence in Education*, 229-238.
- Broadbent, J., & Poon, W. L. (2015). Self-regulated learning strategies & academic achievement in online higher education learning environments: A systematic review. *The Internet and Higher Education*(27), 1-13.
- Cameron, A. C., & Trivedi, P. K. (1998). *Regression analysis of count data*. Cambridge University Press.
- Chen, C. S. (2002). Self-regulated learning strategies and achievement in an introduction to information systems course. *Information Technology, Learning and Performance Journal*.
- Dabbagh, N., & Kitsantas, A. (2004). Supporting self-regulation in student-centered web-based learning environments. *International Journal on E-learning*, 40-47.
- Dabbagh, N., & Kitsantas, A. (2005). Using web-based pedagogical tools as scaffolds for self-regulated learning. *Instructional Science*, 513-540.
- Freitas, A. P., & Rodrigues, S. G. (2005). A avaliação da confiabilidade de questionário: uma análise utilizando o coeficiente alfa de Cronbach. XII SIMPEP.
- Mato, C. A., & Veiga, R. T. (2000). Avaliação da qualidade percebida de serviços: um estudo em uma organização não-governamental. São Paulo: Caderno de Pesquisa em Administração.
- Miguel, P. C., & Salomi, G. E. (2004). Uma revisão dos modelos de medição de qualidade em serviços. Rio de Janeiro: Revista Produção.
- Milan, G. S., & Trez, G. (2005). Pesquisa de satisfação: um modelo para planos de saúde. *RAE Eletrônica, Revista de Administração de Empresas*.

- Nussbaumer, A., Hillemann, E. C., Gütl, C., & Albert, D. (2015). A competence-based service for supporting self-regulated learning in virtual environments. *Journal of Learning Analytics*, 101-133.
- Oviedo, H. C., & Campo-Arias, A. (2005). Aproximación al uso del coeficiente alfa de Cronbach. *Revista Colombiana de Psiquiatría*.
- Panadero, E., Kirschner, P. A., Järvelä, S., Malmberg, J., & Järvenoja, H. (2015). How Individual Self-Regulation Affects Group Regulation and Performance A Shared Regulation Intervention. *Small Group Research*.
- Pintrich, P. R. (1999). The role of motivation in promoting and sustaining self-regulated learning. *International Journal of Educational Research*.
- Sabourin, J. L., Mott, B. W., & Lester, J. C. (2012). Early Prediction of Student Self-Regulation Strategies by Combining Multiple Models. *International Educational Data Mining Society*.
- Salomi, G. E., Miguel, P. C., & Abackerli, A. J. (2005). SERVQUAL x SERVPERF: comparação entre instrumentos para avaliação da qualidade de serviços internos. *Gestão & Produção*.
- Sanchez-Santillan, M., Paule-Ruiz, M., Cerezo, R., & Alvarez-García, V. (2016). MeL: Modelo de adaptación dinámica del proceso de aprendizaje en eLearning. *anales de psicología*, 106-114.
- Schoor, C., & Bannert, M. (2012). Exploring regulatory processes during a computer-supported collaborative learning task using process mining. *Computers in Human Behavior*, 1321-1331.
- Schunk, D. H. (2005). Self-regulated learning The educational legacy of Paul R. Pintrich. *Educational Psychologist*.
- Schunk, D. H., & Zimmerman, B. J. (1998). *Self-regulated learning: From teaching to self-reflective practice*. Guilford Press.
- Segedy, J. R., Kinnebrew, J. S., & Biswas, G. (2015). Using coherence analysis to characterize self-regulated learning behaviours in open-ended learning environments. 13-48.
- Sonnenberg, C., & Bannert, M. (2015). Discovering the Effects of Metacognitive Prompts on the Sequential Structure of SRL-Processes Using Process Mining Techniques. *Journal of Learning Analytics*, 72-100.
- Urdan, A. T. (2001). Qualidade de Serviços médicos na perspectiva do cliente. *Revista de Administração de Empresas*.
- Winne, P. H., & Jamieson-Noel, D. (2002). Exploring students' calibration of self reports about study tactics and achievement. *Contemporary Educational Psychology*, 551–572.
- Winne, P. H., & Perry, N. E. (2000). Measuring self-regulated learning. *Handbook of self-regulation*, 531–566.
- You, J. W. (2016). Identifying significant indicators using LMS data to predict course achievement in online learning. *The Internet and Higher Education*, 23-30.
- Yukselturk, E., & Bulut, S. (2009). Gender differences in self-regulated online learning environment. *Journal of Educational Technology & Society*, 12-22.
- Zimmerman, B. J., & Martinez-Pons, M. (1988). Construct validation of a strategy model of student self-regulated learning. *Journal of Educational Psychology*.

SOBRE O ORGANIZADOR

Everson Mario Novak Possui graduação em Tecnologia em Sistemas para Internet, Especialização em Desenvolvimento Web e MBA em Gestão de TI pela Faculdade Educacional de Ponta Grossa (Faculdade UNIÃO). Atualmente está cursando Mestrado em Informática na PUCPR - Pontifícia Universidade Católica do Paraná é professor do curso de Sistemas de Informação na Faculdades Integradas de Itararé – FAFIT. Ainda como Professor pela PUCPR na TECPUC na unidade de Ponta Grossa. É Analista de Sistemas, programador e tem experiência na área de Ciência da Computação, com ênfase em Arquitetura de Sistemas de Computação, Agentes de Software e Inteligência artificial.

Agência Brasileira do ISBN

ISBN 978-85-85107-14-7



9 788585 107147