UNIVERSIDADE DE SÃO PAULO

Instituto de Ciências Matemáticas e de Computação

Uma arquitetura de referência orientada a serviços para o desenvolvimento de ambientes de aprendizagem ubíqua

Átila Rabelo Lopes

Tese de Doutorado do Programa de Pós-Graduação em Ciências de Computação e Matemática Computacional (PPG-CCMC)

SERVIÇO DE PÓS-GRADUAÇÃO DO ICMC-USP

Data de Depósito:

Assinatura: _____

Átila Rabelo Lopes

Uma arquitetura de referência orientada a serviços para o desenvolvimento de ambientes de aprendizagem ubíqua

Tese apresentada ao Instituto de Ciências Matemáticas e de Computação – ICMC-USP, como parte dos requisitos para obtenção do título de Doutor em Ciências – Ciências de Computação e Matemática Computacional. *VERSÃO REVISADA*

Área de Concentração: Ciências de Computação e Matemática Computacional

Orientadora: Profa. Dra. Rosana Teresinha Vaccare Braga

Coorientadora: Profa. Dra. Ellen Francine Barbosa

USP – São Carlos
Maio de 2021

Ficha catalográfica elaborada pela Biblioteca Prof. Achille Bassi
e Seção Técnica de Informática, ICMC/USP,
com os dados inseridos pelo(a) autor(a)

L864a Lopes, Átila Rabelo
 Uma arquitetura de referência orientada a
serviços para o desenvolvimento de ambientes de
aprendizagem ubíqua / Átila Rabelo Lopes;
orientadora Rosana Teresinha Vaccare Braga;
coorientadora Ellen Francine Barbosa. -- São
Carlos, 2021.
 206 p.

 Tese (Doutorado - Programa de Pós-Graduação em
Ciências de Computação e Matemática Computacional) --
Instituto de Ciências Matemáticas e de Computação,
Universidade de São Paulo, 2021.

 1. Arquitetura de Referência. 2. Arquitetura
Orientada a Serviços (SOA). 3. Aprendizagem Ubíqua (u-
learning). 4. Ferramenta de Autoria. 5. Sistemas
Sensíveis ao Contexto (CSS). I. Braga, Rosana
Teresinha Vaccare , orient. II. Barbosa, Ellen
Francine, coorient. III. Título.

Átila Rabelo Lopes

**A service oriented reference architecture for the
development of ubiquitous learning environments**

Doctoral dissertation submitted to the Instituto de
Ciências Matemáticas e de Computação – ICMC-
USP, in partial fulfillment of the requirements for the
degree of the Doctorate Program in Computer Science
and Computational Mathematics. *FINAL VERSION*

Concentration Area: Computer Science and
Computational Mathematics

Advisor: Profa. Dra. Rosana Teresinha Vaccare Braga
Co-advisor: Profa. Dra. Ellen Francine Barbosa

USP – São Carlos
May 2021

Dedico este trabalho aos meus pais, Bernardo Martins Lopes e Zélia Rabelo Lopes, que sempre me apoiaram e se dedicaram para que eu pudesse alcançar meus sonhos e chegar até aqui.

AGRADECIMENTOS

Gostaria de agradecer, primeiramente a Deus, que sempre está ao meu lado e protegeu minha família quando eu não pude estar por perto.

Meu agradecimento especial é para minha esposa, Mônica A. Veras, e meus filhos, Bernardo Martins Lopes Neto e Moana Alves Veras Lopes, por todo sacrifício que fizeram para que eu pudesse fazer este curso de doutorado e por confiarem em mim. Amo vocês!

Também agradeço aos meus amigos, familiares e professores da USP, em especial à minha orientadora, Rosana Vaccare Braga, e co-orientadora, Ellen Francine Barbosa, por todo conhecimento e dedicação que me deram.

Por fim, agradeço aos colegas de trabalho, professores de computação da UESPI, por me ajudarem durante todo o período deste curso, e aos meus alunos que colaboraram com esta pesquisa.

*“A educação não transforma o mundo.
Educação muda as pessoas.
Pessoas transformam o mundo.”
(Paulo Freire)*

RESUMO

LOPES, A. R. **Uma arquitetura de referência orientada a serviços para o desenvolvimento de ambientes de aprendizagem ubíqua**. 2021. 206 p. Tese (Doutorado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2021.

Ambientes de aprendizagem ubíqua ou *u-learning* (*ubiquitous learning*) representam a nova geração dos ambientes virtuais de aprendizagem (AVA), sendo aplicados no domínio da educação como recurso didático capaz de oferecer diversos benefícios para o processo do ensino-aprendizagem, tais como aprendizagem com mobilidade, recomendação de conteúdos personalizados e adaptação proativa dos materiais didáticos ao contexto da aprendizagem do aluno, por meio dos dispositivos e tecnologias da computação móvel, sensível ao contexto e redes de comunicação sem fio. No entanto, o desenvolvimento de ambientes *u-learning* é uma tarefa difícil e demorada, ainda mais quando o ambiente incorpora serviços de autoria de conteúdos de aprendizagem sensíveis ao contexto, que aumenta a complexidade do desenvolvimento. Motivado por este cenário, esta tese de doutorado investigou e definiu uma arquitetura de referência orientada a serviços para o desenvolvimento de ambientes de aprendizagem ubíqua, denominada MOA-RA, tendo como objetivo principal contribuir para projetos arquiteturais de ambientes *u-learning*. A arquitetura foi construída com base no processo de desenvolvimento de arquiteturas de referência “ProSA-RA” e validada por meio de três tipos de avaliação arquitetural: Inspeção por *checklist*, questionário com especialistas do domínio e estudos de caso como prova de conceito da instanciação da arquitetura proposta. Os resultados foram positivos e indicaram que a MOA-RA atingiu os objetivos esperados.

Palavras-chave: Arquitetura de Referência, Arquitetura Orientada a Serviços (SOA), Aprendizagem Ubíqua (*u-learning*), Ferramenta de Autoria, Sistemas Sensíveis ao Contexto (CSS).

ABSTRACT

LOPES, A. R. **A service oriented reference architecture for the development of ubiquitous learning environments**. 2021. 206 p. Tese (Doutorado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2021.

Ubiquitous learning environments or u-learning represent the new generation of virtual learning environments (VLE), applied in the field of education as a didactic resource capable of offering several benefits for the teaching-learning process, such as mobile learning, recommendation of personalized content and proactive adaptation of didactic materials to the student's learning context, through the devices and technologies of mobile computing, context-sensitive and wireless communication networks. However, the development of u-learning environments is a difficult and time-consuming task, especially when the environment incorporates context-sensitive learning content authoring services, which increases the complexity of development. Motivated by this scenario, this doctoral thesis investigated and defined a service-oriented reference architecture for the development of ubiquitous learning environments, called MOA-RA, with the main objective of contributing to architectural projects of u-learning environments. The architecture was built based on the "ProSA-RA" reference architecture development process and validated through three types of architectural assessment: Inspection by checklist, questionnaire with domain experts and case studies as proof of concept of the instantiation of the proposed architecture. The results were positive and indicated that the MOA-RA achieved the expected objectives.

Keywords: Reference Architecture, Service Oriented Architecture (SOA), Ubiquitous Learning (u-learning), Authoring Tool, Context Sensitive Systems (CSS).

LISTA DE ILUSTRAÇÕES

Figura 1 – Relação entre modelo de referência, estilo e arquitetura de referência	31
Figura 2 – Visão geral do processo ProSA-RA	34
Figura 3 – Estrutura básica da SOA	36
Figura 4 – Fases do <i>M-learning</i>	42
Figura 5 – Relacionamento entre unidades de aprendizagem	49
Figura 6 – Associação entre unidade e informação do contexto	49
Figura 7 – Camadas conceituais da plataforma OPPIA	50
Figura 8 – Arquitetura do ALESS	52
Figura 9 – Visão geral da <i>Ref-mLearnig</i>	53
Figura 10 – Arquitetura das aplicações móveis	56
Figura 11 – Camadas do “ <i>CROP Object</i> ”	58
Figura 12 – Arquitetura SOA para aprendizagem sensível ao contexto	59
Figura 13 – Visão geral do Midga	60
Figura 14 – Visão geral do ambiente UFC- <i>Inventor</i>	61
Figura 15 – Conceitos-chave do grupo GEE	74
Figura 16 – Modelo estrutural da arquitetura	87
Figura 17 – Visão geral da MOA-RA	90
Figura 18 – Visão ampliada dos Módulos Educacionais (MOA-RA)	91
Figura 19 – Visão de Módulos da MOA-RA	97
Figura 20 – Visão em Tempo de Execução (MOA-RA)	99
Figura 21 – Visão de Implantação (MOA-RA)	101
Figura 22 – Tipo de experiência com arquitetura de software (arquiteto)	114
Figura 23 – Tipo de experiência com ambientes educacionais (esp. do domínio)	115
Figura 24 – Tempo de experiência dos especialistas	115
Figura 25 – Resultado quantitativo (“Aspectos gerais” – arquitetos)	117
Figura 26 – Resultado quantitativo (“Aspectos arquiteturais” - arquiteto)	118
Figura 27 – Resultado avaliação quantitativa (Arquitetos)	120
Figura 28 – Resultado Avaliação com Arquitetos (grupo de aspectos)	120
Figura 29 – Comparativo dos Resultados “Aspectos gerais” (arquitetos e esp. domínio)	122
Figura 30 – Resultado - “Aspectos domínio educacional” (quantitativo)	123
Figura 31 – Resultado da avaliação dos especialistas do domínio (qualitativo)	124
Figura 32 – Tabuleiro do “ <i>Battleship</i> ”	129
Figura 33 – Processo de Especialização Arquitetural	131

Figura 34 – Módulos da instanciação arquitetural do cenário-1	132
Figura 35 – Instanciação parcial da arquitetura do cenário-1	133
Figura 36 – Instanciação total da arquitetura do cenário-2	138
Figura 37 – Protótipo do jogo Batalha Naval: Telas Gerais	139
Figura 38 – Funcionalidades do jogo (Pré-batalha)	140
Figura 39 – Telas de batalha do jogo	141
Figura 40 – Protótipo do <i>Class Path</i> (plataforma de gerenciamento)	143
Figura 41 – Protótipo do aplicativo “ <i>App-Author</i> ”	144
Figura 42 – Protótipo do aplicativo <i>App-Learn</i> (plataforma de aprendizagem)	145
Figura 43 – Comparativo entre os protótipos dos estudos de caso (motivação, usabilidade e percepção)	152

LISTA DE TABELAS

Tabela 1 – Comparativo das arquiteturas dos trabalhos relacionados	63
Tabela 2 – Conceitos do grupo ULE	70
Tabela 3 – Comparativo dos conceitos GEE	74
Tabela 4 – Relação entre conceitos e requisitos (SOA)	77
Tabela 5 – Comparativo dos conceitos SOA	79
Tabela 6 – Mapeamento parcial dos requisitos de sistema, conceitos requisitos arquitet- urais (GEE)	80
Tabela 7 – Mapeamento parcial dos requisitos de sistema, conceitos e requisitos arquit- eturais (ULE)	82
Tabela 8 – Mapeamento dos requisitos de sistema, conceitos e requisitos arquitet- urais SOA	85
Tabela 9 – Pacotes e sub-pacotes da visão de Módulos (MOA-RA)	95
Tabela 10 – Resultado da avaliação por inspeção (<i>Checklist</i>)	107
Tabela 11 – Materiais da avaliação com especialistas	112
Tabela 12 – Escala das respostas e resultados da avaliação da MOA-RA	113
Tabela 13 – Resultado da avaliação com arquitetos	116
Tabela 14 – Resultado da avaliação com especialistas do domínio educacional	121
Tabela 15 – Mapeamento dos requisitos arquitet-urais (cenário-1)	132
Tabela 16 – Mapeamento dos requisitos arquitet-urais (cenário-2)	136
Tabela 17 – Resultados das avaliações dos protótipos dos estudos de caso	150

SUMÁRIO

1	INTRODUÇÃO	23
1.1	Contextualização e Motivações	23
1.2	Hipótese e Questões de Pesquisa	26
1.3	Objetivos	26
1.4	Organização do Trabalho	27
2	FUNDAMENTAÇÃO TEÓRICA	29
2.1	Consideração Iniciais	29
2.2	Arquitetura de <i>Software</i> e Terminologia Associada	29
2.2.1	<i>Arquiteturas de Referência e ProSA-RA</i>	31
2.2.1.1	<i>Processo ProSA-RA</i>	33
2.2.2	<i>Arquitetura Orientada a Serviços (SOA)</i>	35
2.3	Computação Ubíqua	40
2.4	Aprendizagem Móvel e Ubíqua	42
2.5	Considerações Finais	45
3	ARQUITETURAS DE AMBIENTES DE APRENDIZAGEM MÓ- VEL E UBÍQUA	47
3.1	Considerações Iniciais	47
3.2	Arquiteturas <i>M-learning</i> e <i>U-learning</i>	47
3.3	Análise Comparativa	62
3.4	Considerações Finais	64
4	DESENVOLVIMENTO DA ARQUITETURA DE REFERÊNCIA PRO- POSTA	65
4.1	Considerações iniciais	65
4.2	Investigação das fontes de informação (RA-1)	66
4.2.1	<i>Investigação do Grupo dos Ambientes de Aprendizagem Ubíqua</i>	67
4.2.2	<i>Investigação do Grupo dos Ambientes Educacionais Genéricos (GEE)</i>	71
4.2.3	<i>Investigação do Grupo de Arquitetura Orientada a Serviços (SOA)</i>	75
4.3	Estabelecimento dos Requisitos Arquiteturais (RA-2)	79
4.3.1	<i>Requisitos Arquiteturais do grupo GEE (ambientes educacionais genéricos)</i>	80
4.3.2	<i>Requisitos Arquiteturais do grupo ULE (ambientes u-learning)</i>	82

4.3.3	<i>Requisitos Arquiteturais do grupo SOA</i>	84
4.4	<i>Projeto da Arquitetura de Referência (RA-3)</i>	86
4.4.1	<i>Visão Geral</i>	88
4.4.2	<i>Visão de Módulos</i>	93
4.4.3	<i>Visão em Tempo de Execução</i>	98
4.4.4	<i>Visão de Implantação</i>	100
4.5	<i>Avaliação da Arquitetura de Referência (RA-4)</i>	102
4.6	<i>Considerações Finais</i>	102
5	AVALIAÇÃO DA ARQUITETURA DE REFERÊNCIA MOA-RA . .	105
5.1	<i>Considerações iniciais</i>	105
5.2	<i>Avaliação por Inspeção</i>	106
5.2.1	<i>Análise dos resultados da avaliação (inspeção)</i>	108
5.3	<i>Avaliação com Especialistas</i>	110
5.3.1	<i>Avaliadores (grupos de especialistas)</i>	110
5.3.2	<i>Questionários e escalas</i>	111
5.3.3	<i>Perfil dos avaliadores</i>	113
5.3.4	<i>Resultados da avaliação do grupo dos arquitetos</i>	115
5.3.5	<i>Resultados da avaliação do grupo dos especialistas do domínio edu- cacional</i>	120
5.3.6	<i>Riscos e ameaças à validade da avaliação</i>	125
5.4	<i>Considerações Finais</i>	125
6	ESTUDO DE CASO DA MOA-RA	127
6.1	<i>Considerações Iniciais</i>	127
6.2	<i>Cenários de Aplicação da MOA-RA</i>	128
6.2.1	<i>Cenário 1</i>	128
6.2.1.1	<i>Instanciação Parcial da Arquitetura do Cenário-1</i>	130
6.2.2	<i>Cenário 2</i>	134
6.2.2.1	<i>Instanciação Total da Arquitetura do Cenário-2</i>	135
6.3	<i>Implementação dos protótipos educacionais dos cenários da MOA-RA</i>	137
6.3.1	<i>Protótipo do jogo Batalha Naval (cenário-1)</i>	137
6.3.2	<i>Protótipo do Ambiente de Aprendizagem Ubíqua (cenário-2)</i>	142
6.4	<i>Estudos de Caso com os protótipos da MOA-RA</i>	146
6.4.1	<i>Execução dos Estudos de Caso</i>	147
6.4.2	<i>Análise dos Resultados</i>	149
6.4.2.1	<i>Resultados do Estudo 1</i>	150
6.4.2.2	<i>Resultados Estudo 2</i>	151
6.5	<i>Considerações Finais</i>	153

7	CONCLUSÃO	155
7.1	Considerações Iniciais	155
7.2	Visão Geral do trabalho	155
7.3	Contribuições	157
7.4	Dificuldades e Limitações	158
7.5	Trabalhos Futuros	158
7.6	Publicações Científicas da Pesquisa	159
REFERÊNCIAS		161
APÊNDICE A	AVALIAÇÃO - ESPECIALISTA ARQUITETO	167
APÊNDICE B	AVALIAÇÃO - ESPECIALISTA DO DOMÍNIO EDUCACIONAL	187
APÊNDICE C	AVALIAÇÃO - PROTÓTIPO BATALHA NAVAL	199
APÊNDICE D	AVALIAÇÃO - PROTÓTIPO CLASS PATH	201
APÊNDICE E	AVALIAÇÃO - PROTÓTIPO APP-AUTHOR	203
APÊNDICE F	AVALIAÇÃO - PROTÓTIPO APP-LEARN	205

INTRODUÇÃO

Este capítulo apresenta uma visão geral sobre esta tese de doutorado, mostrando o contexto no qual está inserida, principais motivações, hipótese e questões de pesquisa, e os objetivos a serem alcançados. Ao final, é apresentada a organização dos demais capítulos deste trabalho.

1.1 Contextualização e Motivações

Aprendizagem ubíqua ou *u-learning* (*ubiquitous learning*) é um campo de pesquisa emergente, que consiste no uso das tecnologias e recursos da computação ubíqua aplicado na educação para favorecer o processo de ensino-aprendizagem, provendo serviços e materiais didáticos que podem se adaptar ao contexto da aprendizagem do aluno, em qualquer lugar e a qualquer hora. “Um dos principais objetivos da aprendizagem ubíqua é fornecer informações adequadas e oportunas no lugar e na hora certa” (HWANG *et al.*, 2011).

Os ambientes *u-learning* representam a nova geração da aprendizagem eletrônica ou *e-learning* (*eletronic learning*), definindo um novo modelo de ambientes virtuais de aprendizagem integrados às novas tecnologias da informação e comunicação (NTICs). Assim, *smartphones* e *tablets* equipados com dispositivos de geolocalização (GPS - *Global Positioning System*), sensores, conexão à internet e redes de comunicação sem fio, oferecem novos serviços e possibilidades que permitem expandir os horizontes da educação, dando origem a novos modelos de *e-learning* como: (i) *M-learning* (*mobile learning* ou aprendizagem móvel); (ii) *Context-aware learning* (aprendizagem sensível ou ciente do contexto); e (iii) *u-learning*, que é a junção dos dois modelos anteriores (LOPES *et al.*, 2017), ou seja, herda de *M-learning* o uso de tecnologias da computação móvel e de *Context-aware learning* a capacidade de obter e gerenciar informações de contexto do aluno.

Na educação, contexto pode ser uma informação sobre o estilo de aprendizagem do

aluno, atividades correntes, local e hora onde a aprendizagem ocorre, tipo de dispositivo utilizado pelo aluno, preferências de estudo, entre outros (YU; ZHOU; SHU, 2010).

Os ambientes *u-learning* podem trazer benefícios e vantagens para o processo de ensino e aprendizagem, fornecendo serviços personalizados às características individuais do aluno, mobilidade dos recursos didáticos e, com isso, favorecer a aprendizagem no momento e local adequado (*Just-in-time*), fornecer conteúdos e objetos de aprendizagem (OA) adaptados ao contexto do usuário, entre outros benefícios educacionais (MARCOLINO, 2019; FILHO, 2016; CHIANG; YANG; HWANG, 2014).

Na literatura, existem várias propostas de ambientes *u-learning* como: **Edubi-Mobile** (PINHEIRO, 2019); **ALESS** (HSU *et al.*, 2015); **OPPIA** (BRAVO *et al.*, 2016); **Youubi** (MONTEIRO *et al.*, 2015); **GlobalEdu** (BARBOSA, 2007); entre outros. No entanto, o reúso dos ambientes já existentes pode não ser tão simples assim, exigindo planejamentos e adequações do professor para utilizar recursos didáticos que não foram construídos de acordo com suas necessidades e objetivos de ensino (FILHO, 2016).

Neste cenário, as ferramentas de autoria *u-learning* podem ajudar a contornar os problemas apresentados, em relação ao reúso e desenvolvimento de novos ambientes *u-learning*, pois são ferramentas cuja proposta principal é simplificar o processo de desenvolvimento de aplicações e conteúdos de aprendizagem ubíqua, fornecendo mecanismos para que usuários (professor) sem conhecimentos avançados em informática possam criar seus próprios recursos digitais (FLÔRES; TARROUCO; REATEGUI, 2011).

Em geral, para permitir a construção de aplicações *u-learning* integradas aos dispositivos com tecnologias sensíveis ao contexto (GPS, sensor, câmera, etc.), sem a necessidade do usuário implementar o código da aplicação, as ferramentas de autoria investigadas neste trabalho utilizam mecanismos de geração automática do código-fonte da aplicação que deve ser instalada no dispositivo móvel do usuário final, a exemplo das seguintes ferramentas de autoria: **UFC-Inventor** (FILHO, 2016); **Tangram** (MARCOLINO, 2019); **UoLmP** (GÓMEZ *et al.*, 2014); **MAT for ARLearn** (TABUENCA *et al.*, 2014); **Midgar** (GARCÍA *et al.*, 2014); **APP Inventor** (MIT, 2012); **Lemonade** (GIEMZA; BOLLEN; HOPPE, 2011).

Apesar das facilidades e benefícios oferecidos por essas ferramentas, o uso de mecanismos para a modelagem e geração automática do código pode ocasionar problemas de inconsistência no conteúdo das aplicações criadas, ou seja, qualquer alteração no corpo do código-fonte do aplicativo (conteúdo, funcionalidade, *layout* da tela) não é atualizada na versão do aplicativo que já está instalado no dispositivo, pois gera um novo código-fonte diferente do anterior (BRAMBILLA; CABOT; WIMMER, 2012). Este problema obriga o usuário a ter que desinstalar a versão antiga e instalar a versão atualizada do aplicativo (com as alterações), sempre que houver alguma modificação necessária.

Além disso, apesar de facilitar bastante o processo de construção de aplicativos mó-

veis, algumas ferramentas de autoria, como no caso do “*APP Inventor*”, ainda exigem do autor certo conhecimento em lógica de programação e banco de dados para criar aplicações sensíveis ao contexto mais completas. Para tornar esse quadro ainda mais grave, a variedade de sistemas operacionais para dispositivos móveis (*Android, iOS, Windows Phone, BlackBerry*, etc.) é uma limitação para a maioria das ferramentas de autoria que geram aplicações para sistemas operacionais específicos (FILHO, 2016).

Portanto, o desenvolvimento de ambientes de autoria *u-learning* para dispositivos móveis é uma tarefa difícil e demorada, que envolve várias questões complexas como: adaptação às constantes mudanças do contexto; quantidade de informações do contexto que o ambiente deve tratar; geração automática do código; comunicação com dispositivos de tecnologias heterogêneas, de *hardware* e *software* (PINHEIRO, 2019; DUARTE FILHO, 2016). Além disso, essa tarefa é feita sem o apoio arquitetural adequado que possa ajudar a superar alguns dos desafios identificados neste processo (FILHO, 2016).

Desta forma, é importante definir uma arquitetura de referência específica para o domínio em questão, que forneça as diretrizes necessárias para conduzir o desenvolvimento de ambiente de autoria *u-learning* desta natureza (MACHADO, 2015).

Arquitetura de referência é um tipo de arquitetura de *software* para apoiar projetos arquiteturais de um determinado domínio de aplicação, servindo como modelo para o desenvolvimento de arquiteturas concretas de um conjunto de sistemas do mesmo domínio (NAKAGAWA; FLÁVIO; BECKER, 2012; SHAW; GARLAN, 1996). Além de contribuir para a redução do tempo e esforços necessários para o desenvolvimento de *software*, as arquiteturas de referência também oferecem outros benefícios importantes para projetos arquiteturais, facilitando a compreensão do domínio de aplicação, reúso, manutenção e evolução do sistema (NAKAGAWA, 2006; BASS; CLEMENTS; KAZMAN, 2003).

Considerando a importância do apoio arquitetural para o desenvolvimento de aplicações *u-learning*, que possam estar imersas num ambiente tecnologicamente heterogêneo, interagindo com aplicações de diferentes linguagens de programação e plataformas de *hardware*, o paradigma da arquitetura orientada a serviços ou SOA (*Service-Oriented Architecture*) mostra-se apropriada para tratar desses aspectos, além de favorecer questões do reúso, interoperabilidade entre aplicações desenvolvidas em diferentes plataformas de *hardware* e *software*, entre outros, (ANGELOV; TRIENEKENS; GREFEN, 2008).

SOA é um estilo arquitetural para apoiar a construção e integração de sistemas, no qual as aplicações são estruturadas em elementos modulares que representam a unidade básica da arquitetura, chamados de serviços. Um serviço *web* consiste numa funcionalidade simples e específica de um sistema, publicado em um repositório *web* para ser localizado e utilizado por diferentes aplicações e serviços (ERL, 2005).

Outra característica importante da SOA é o conceito de separação de assuntos, que

consiste em dividir um problema grande em partes menores, para que a solução também possa ser decomposta em várias partes, onde cada parte deve solucionar um determinado assunto. Essa característica é baseada nos princípios de reúso, baixo acoplamento e abstração (ERL, 2005).

Por fim, vale ressaltar que foram encontrados alguns trabalhos relacionados à arquitetura de ambientes de autoria de aplicações *m-learning* e *u-learning*, tais como os trabalhos de Pinheiro (2019), Marcolino (2019), Filho (2016), Escudero *et al.* (2018), Bravo *et al.* (2016), Moebert, Höfler e Lucke (2016), Monteiro *et al.* (2015), Zhao e Wang (2011), Gómez *et al.* (2014), García *et al.* (2014), Tabuenca *et al.* (2014). Porém, a maioria apresenta apenas a arquitetura concreta dos ambientes propostos nos respectivos trabalhos, e as poucas arquiteturas de referência encontradas durante as pesquisas são baseadas nas técnicas do Desenvolvimento Dirigido por Modelos (*Model-Driven Development* - MDD), que por envolver mecanismos menos intuitivos para a maioria dos programadores, são difíceis de serem utilizados, aumentando a complexidade do desenvolvimento (FILHO, 2016).

1.2 Hipótese e Questões de Pesquisa

Diante do cenário apresentado, esta tese de doutorado apresenta a seguinte hipótese:

“Baseado nas características e princípios da SOA, é possível definir uma arquitetura de referência para o desenvolvimento de ambientes de autoria u-learning, sem a necessidade de utilizar mecanismos de geração automática do código-fonte para criar conteúdos de aprendizagem sensíveis ao contexto, para plataforma dos dispositivos móveis.”

A partir desta hipótese, foram levantadas as seguintes questões de pesquisa:

- Como professores sem conhecimento em programação podem construir conteúdos sensíveis ao contexto para a plataforma dos dispositivos móveis?
- Como uma ferramenta de autoria pode criar atividades *u-learning* para dispositivos móveis, sem utilizar mecanismos de geração automática do código da atividade criada?
- Como garantir a persistência do conteúdo de uma atividade sensível ao contexto, alterado após sua entrega?

1.3 Objetivos

Motivado por este cenário e considerando as questões levantadas, o principal objetivo desta tese é contribuir para projetos arquiteturais de ambientes de aprendizagem ubíqua, por meio de investigação e definição de uma arquitetura de referência orientada a serviços para apoiar o desenvolvimento de ambientes de autoria de conteúdos de aprendizagem sensíveis ao contexto,

para plataforma dos dispositivos móveis. A arquitetura proposta chama-se MOA-RA¹, e é voltada diretamente a arquitetos de *software* e desenvolvedores de sistemas *u-learning*. De maneira indireta, contribui-se também para os usuários desses ambientes educacionais (professores, alunos e gestores de escolas), pois por ser baseada no princípio de reuso a arquitetura favorece a construção de ambientes de qualidade e que podem evoluir com facilidade.

Para atingir o objetivo geral da pesquisa, alguns objetivos específicos são considerados ao longo deste trabalho, entre eles:

- Identificar os requisitos, tecnologias e características relacionadas aos ambientes de aprendizagem ubíqua, investigando trabalhos científicos por meio de revisões e mapeamentos sistemáticos da literatura;
- Identificar os problemas relacionados à autoria de conteúdos e aplicações *u-learning*, e as soluções propostas na literatura da área;
- Definir os requisitos arquiteturais e construir a arquitetura de referência MOA-RA, levando em consideração os problemas e soluções encontrados. A arquitetura foi modelada em quatro visões arquiteturais, ilustrando suas características, componentes e relacionamentos, sob diferentes pontos de vista do desenvolvimento;
- Verificar os pontos positivos e negativos da MOA-RA, em relação às questões de qualidade e aplicabilidade prática para validar seu principal objetivo, realizando avaliações formais (quantitativa e qualitativa) e experimentos práticos com a arquitetura;
- Definir diferentes cenários de aprendizagem *u-learning* para a instanciação arquitetural da MOA-RA, e implementação dos respectivos protótipos educacionais, para realizar dois estudos de caso junto a professores e alunos, como prova de conceito da aplicação prática da MOA-RA, em cenários reais de aprendizagem.

1.4 Organização do Trabalho

Neste capítulo, foi feita uma introdução geral da presente tese de doutorado, apresentando uma contextualização sobre os principais aspectos das áreas envolvidas no trabalho, motivações, hipótese, questões de pesquisa e objetivos da tese. O restante do trabalho está organizado da seguinte forma: O Capítulo 2 apresenta os assuntos das áreas que representam a base teórica deste trabalho de tese, abordando os principais conceitos, princípios e características das arquiteturas de referência, SOA, aprendizagem ubíqua e ferramentas de autoria *u-learning*. No Capítulo 3 são apresentados alguns trabalhos relacionados ao domínio desta pesquisa. O Capítulo 4 discute o

¹ A sigla MOA-RA foi criada como uma homenagem pessoal à filha do autor desta tese, “Moana”. Além disso, é uma palavra de origem havaiana que significa “oceano”, “mar profundo”, representando assim as possibilidades de aplicações da arquitetura. “RA” é uma abreviatura de “Reference Architecture”

processo de desenvolvimento da arquitetura proposta, mostrando as etapas que foram seguidas. O Capítulo 5 mostra as avaliações realizadas com a arquitetura MOA-RA, no âmbito deste trabalho. O Capítulo 6 apresenta dois estudos de caso realizados com os protótipos instanciados a partir da arquitetura de referência MOA-RA. Por fim, no Capítulo 7 são apresentadas as conclusões desta tese de doutorado.

FUNDAMENTAÇÃO TEÓRICA

2.1 Consideração Iniciais

Este capítulo apresenta os principais conceitos relacionados aos assuntos que formam a base teórica deste trabalho de tese de doutorado. A Seção 2.2 aborda o tema de arquitetura de *software* e terminologias associadas, focando em arquitetura de referência e processo de desenvolvimento de arquiteturas de referência (Seção 2.2.1), e arquitetura orientada a serviços (Seção 2.2.2). A Seção 2.3 discute sobre a computação ubíqua, incluindo a computação móvel e sensibilidade ao contexto. Na Seção 2.4, são apresentados os conceitos da aprendizagem móvel e ubíqua. Por fim, na Seção 2.5 são apresentadas as considerações finais do capítulo.

2.2 Arquitetura de *Software* e Terminologia Associada

Arquitetura de *software* pode ser definida como a estrutura de um sistema, que mostra a forma de organização dos componentes do *software*, suas propriedades externas, assim como os relacionamentos existentes entre tais componentes, sendo vista como uma abstração de alto nível do sistema. Seu papel principal é diminuir a distância entre requisitos do *software* e sua implementação (BASS; CLEMENTS; KAZMAN, 2003).

Em termos gerais, arquitetura é um artefato que serve para auxiliar o processo de desenvolvimento do sistema, por meio de representações gráficas e informações que descrevem todos os aspectos importantes que devem ser considerados durante a implementação do *software*. Segundo Bass, Clements e Kazman (2003), arquitetura de *software* representa o principal fator para determinar a qualidade do sistema, pois fornece as diretrizes e modelos de boas práticas que favorecem o desenvolvimento de sistemas de qualidade.

Desta forma, a arquitetura de *software* tem um papel fundamental em qualquer projeto de sistema (CLEMENTS; NORTHROP, 2002). A importância das arquiteturas para projetos de

software deve-se aos benefícios que podem oferecer para tais projetos, como: diminuição do tempo de desenvolvimento; redução dos custos e riscos do projeto; possibilidade de reúso de artefatos e códigos; facilidade de evolução do sistema; favorecimento da qualidade do sistema desenvolvido; entre outros (SHAW; GARLAN, 1996).

Existem diversos conceitos e terminologias na área de arquitetura de *software*, que definem um vocabulário comum entre os pesquisadores, por exemplo: instância arquitetural; padrão arquitetural; estilo arquitetural; arquitetura concreta; arquitetura de referência; modelo de referência.

Arquitetura concreta é a arquitetura de um determinado *software* e consiste em um conjunto de especificações e padrões para conduzir o desenvolvimento das partes que compõem esse *software*. Por sua vez, instância arquitetural é o termo que serve como sinônimo de arquitetura concreta, indicando que ela foi concebida de uma arquitetura de referência para um domínio de aplicação específica (NAKAGAWA, 2006).

Padrão arquitetural representa um modelo importante de organização estrutural do sistema, fornecendo um conjunto pré-definido de subsistemas, com as devidas especificações de suas responsabilidades, regras e diretrizes para organizar as relações entre eles (ERL, 2005). Para cada tipo de sistema (sistemas distribuídos, adaptativos, interativos, etc.) existem padrões arquiteturais específicos, por exemplo, o padrão MVC (*Model-View-Controller*) é um dos mais conhecidos para os sistemas interativos. Esse padrão separa a aplicação em três componentes: o Modelo (*Model*) contém as funcionalidades básicas; a Visão (*View*) gerencia as informações exibidas para o usuário; e o Controlador (*Controller*) processa as entradas (eventos), considera o estado atual do cliente e determina o próximo estado do sistema.

Estilo arquitetural é um termo mais abrangente, que descreve as diferentes formas de organização estrutural de *software*, definindo a topologia da arquitetura e o conjunto de restrições para os componentes e conectores, com o objetivo de apoiar os desenvolvedores e arquitetos a determinarem a classe na qual o *software* pertence. De acordo com Bass, Clements e Kazman (2003), existem quatro propriedades dos sistemas definidas pelo estilos arquiteturais, sendo:

- Vocabulário: indica os tipos de componentes e conectores que podem ser utilizados na arquitetura;
- Regras de projeto: determinam as regras de composições de componentes permitidas;
- Interpretação semântica: os componentes restringidos pelas regras de projeto têm um significado bem definido;
- Análise: realizada sobre o sistema desenvolvido em um estilo arquitetural.

Segundo Clements e Northrop (2002), modelo de referência representa a separação das funcionalidades de um sistema em partes menores, que se relacionam cooperativamente em

busca de solução para um determinado problema. Em muitos casos, os problemas são específicos para um determinado domínio. Dessa forma, as soluções podem ser compartilhadas entre os indivíduos pertencentes ao mesmo domínio. Um exemplo típico do modelo de referência é a arquitetura orientada a serviços, que carrega a essência do modelo de referência.

As arquiteturas de referência são projetadas para serem aplicadas em uma área de domínio em particular, atuando como uma espécie de mapeamento dos modelos de referência e permitindo que os elementos de *software* trabalhem de forma cooperativa, na implementação das funcionalidades desse modelo (ERL, 2005). Elas podem ser definidas como uma abstração dos componentes do *software* e seus relacionamentos internos, dando condições para implementar as funcionalidades que foram definidas no modelo de referência e atribuídas às partes do sistema. Arquitetura de referência pode ser vista como um banco de conhecimento para um domínio específico, que permite o reúso arquitetural e representam o mapeamento das funcionalidades (elementos de *software*) definidas no modelo de referência adotado (BASS; CLEMENTS; KAZMAN, 2003).

A Figura 1 mostra o relacionamento básico entre modelo de referência e estilo arquitetural, na composição da arquitetura de referência. As subseções a seguir fornecem mais conceitos sobre arquiteturas de referência e arquiteturas orientadas a serviços, por serem diretamente relacionadas ao contexto desta tese de doutorado.

Figura 1 – Relação entre modelo de referência, estilo e arquitetura de referência

Fonte: Bass, Clements e Kazman (2003)

2.2.1 Arquiteturas de Referência e ProSA-RA

Conforme apresentado anteriormente, arquiteturas de referência são utilizadas para auxiliar o processo de desenvolvimento de arquiteturas concretas, para um domínio específico de aplicação. Sua proposta é facilitar a compreensão do domínio de aplicação do sistema construído, a partir da instanciação da arquitetura concreta, provendo um vocabulário comum e uma estrutura organizada do sistema (SHAW; GARLAN, 1996).

Elas representam uma arquitetura generalizada que abrange um conjunto de características e necessidades individuais de diferentes sistemas, mas comuns a um determinado domínio

de aplicação, tendo um conceito bem parecido ao de um *framework* (arcabouço de aplicação) (ERL, 2005).

Uma das primeiras etapas de um projeto de *software* é a fase de especificação da arquitetura, que gera um conjunto de entradas para as etapas seguintes do projeto. Dessa forma, arquiteturas de referência fornecem um direcionamento mais preciso para a especificação de arquiteturas concretas, servindo como guia para projetos arquiteturais, representando um modelo padronizado que permite a interoperabilidade entre sistemas e facilita a incorporação de novos componentes (MULLER, 2008).

Segundo Angelov, Trienekens e Grefen (2008), arquiteturas de referência representam uma estrutura de apoio para projetos de sistema, que contém um conjunto de especificações de infraestrutura, protocolos de comunicação e definições das interfaces dos diversos componentes do sistema final, contribuindo com facilidades e melhorias para a qualidade, manutenção, reúso e evolução dos sistemas.

Um conceito importante da área da arquitetura de referência é o reúso de artefatos-chave, que representa o princípio básico e fundamental da instância arquitetural (GORTON, 2006). O reúso de *software* consiste em incorporar artefatos que já foram construídos, testados e utilizados por outros sistemas, no desenvolvimento de um novo sistema ou evolução do já existente. Desta forma, o reúso contribui para a redução do tempo de desenvolvimento e favorece o ganho de qualidade. Segundo Duarte Filho (2016), “o uso de arquiteturas de referência testadas é uma maneira eficaz de lidar com os vários requisitos não-funcionais (particularmente os requisitos de qualidade), selecionando arquiteturas de referência existentes, que são conhecidas [...]”.

Existem diferentes tipos de artefatos-chave ou unidades reutilizáveis, variando de um padrão arquitetural a um arcabouço pronto, ou até mesmo um sistema completo. Segundo Angelov, Trienekens e Grefen (2008), o reúso pode acontecer de maneira geral, para uma classe de domínio mais ampla ou limitado a um domínio específico. Dessa forma, o reúso torna a arquitetura de referência um componente da base de ativos reutilizáveis de uma organização, servindo como o marco inicial para a criação da arquitetura concreta. Porém, para efetivar a reutilização de artefatos de *software* é preciso considerá-la em todas as etapas de desenvolvimento de *software*, da fase inicial de análise até a implementação do sistema (ERL, 2005).

No início, o conceito de reúso era limitado ao uso de códigos-fonte, realizada por meio de bibliotecas de rotinas, alimentadas por procedimentos e funções. Com o passar do tempo, e o crescimento do paradigma de orientação a objetos (OO), o escopo das unidades reutilizáveis aumentou, principalmente devido às propriedades das classes que encapsulam funções e dados. Isso permitiu incorporar alguns tipos de serviços *web*, como unidade de reúso. Conforme Gallagher (2000), a unidade reutilizável tornou-se ainda maior, por meio de componentes de *software* que descrevem ou realizam uma função específica, devidamente documentada e com interfaces bem definidas. Arquiteturas de referência fazem parte desse aumento no escopo do reúso, pois elevam a unidade reutilizável ao nível arquitetural.

Por outro lado, a literatura aponta alguns problemas relacionados à construção de arquiteturas de referência. Para [Angelov, Trienekens e Grefen \(2008\)](#), propor arquiteturas de referência para um determinado domínio não é uma tarefa fácil, pois requer conhecimento profundo sobre o domínio em questão. A falta de ferramentas para automatizar este processo, a dificuldade de generalizar os aspectos de um domínio, a partir de sistemas existentes, e a carência de técnicas e métodos específicos para representação das arquiteturas, são algumas limitações para a construção de novas arquiteturas de referência ([NAKAGAWA, 2006](#)).

Além disso, documentar uma arquitetura de referência pode ser uma tarefa difícil, devido principalmente pela falta de um padrão universal para a documentação arquitetural, dificuldades em criar e manter atualizados a documentação de sistemas grandes, complexos e evolutivos ([GORTON, 2006](#)). Nesse sentido, existem processos que auxiliam na criação de arquiteturas de referência, como por exemplo, o ProSA-RA, resumido a seguir.

2.2.1.1 Processo ProSA-RA

O desenvolvimento de arquiteturas de referência é um processo difícil e deve ser executado de forma sistemática, seguindo uma metodologia apropriada para o domínio em questão ([NAKAGAWA; SASAKI; MALDONADO, 2009](#)). Existem vários trabalhos propostos na literatura para apoiar o desenvolvimento sistemático de arquiteturas de referência, em diferentes domínios de aplicação como os trabalhos de [Nakagawa \(2006\)](#), [Muller \(2008\)](#), [Bayer et al. \(2004\)](#), dentre outros.

Entre estas propostas, o ProSA-RA ([NAKAGAWA, 2006](#)) foi o processo adotado para auxiliar a construção da arquitetura MOA-RA. A escolha foi motivada por se tratar de um processo simples, eficiente e já consolidado na literatura, além de oferecer flexibilidade para o desenvolvimento de arquiteturas em diferentes domínios de aplicação ([DUARTE FILHO, 2016](#)).

ProSA-RA é um processo de desenvolvimento sistemático, incremental e evolutivo, para a construção de novas arquiteturas de referência, que integra o processo ProSA (*Process based on Software Architecture*), processo iterativo que incorpora as fases de estabelecimento da arquitetura de referência (ProSA-RA), sua especialização (ProSA-S) e instanciação (ProSA-I), fornecendo as diretrizes específicas de cada fase.

A Figura 2 ilustra o processo ProSA-RA, composto por quatro etapas (passos RA) ordenadas da seguinte forma:

- **RA-1** (investigação e seleção de fontes de informação): consiste na identificação e seleção das fontes relacionadas ao domínio de aplicação da arquitetura, com o objetivo de obter o embasamento teórico, conhecer o estado da arte e levantar informações importantes que possam contribuir para o desenvolvimento da arquitetura. As fontes podem ser publicações científicas, livros, especialistas da área (entrevista ou formulários), trabalhos acadêmicos (tese, dissertação), entre outros;

Figura 2 – Visão geral do processo ProSA-RA

Fonte: Nakagawa (2006)

- RA-2** (estabelecimento dos requisitos arquiteturais): os requisitos arquiteturais são estabelecidos com base nas informações obtidas das fontes e também com ajuda das técnicas e ferramentas de levantamento de requisitos, da engenharia de *software*. Primeiro, são identificados os requisitos de sistema do domínio, que serão mapeados para os requisitos da arquitetura de referência e seus respectivos conceitos do domínio. Essa etapa é importante para o estabelecimento da arquitetura, uma vez que requisitos mal definidos ameaçam todo o projeto;
- RA-3** (projeto da arquitetura de referência): corresponde à etapa do estabelecimento da arquitetura, onde são analisados e selecionados os padrões e estilos arquiteturais mais adequados para o domínio e é proposta a arquitetura de referência. Também são definidos nesta etapa todos os aspectos incorporados na arquitetura (elementos, módulos, mecanismos de comunicação, segurança, etc.), levando em consideração os requisitos definidos. Por fim, a arquitetura é modelada em um conjunto de visões arquiteturais (visão geral, visão de módulos, visão em tempo de execução, etc.) e é elaborada a documentação da arquitetura;
- RA-4** (avaliação da arquitetura de referência): tem como objetivo identificar e corrigir as falhas e inconsistências presentes na arquitetura recém-construída. A avaliação pode ser feita utilizando diferentes técnicas e instrumentos de avaliação arquitetural, tais como: inspeção por *checklist*; questionário com especialistas do domínio; instanciação arquitetural com implementação de protótipos; estudo de caso; entre outros.

2.2.2 Arquitetura Orientada a Serviços (SOA)

As constantes mudanças de negócios e serviços empresariais exigem sistemas flexíveis e aptos para permitir alterações, inserir novas funcionalidades e tecnologias computacionais. Também é necessário facilitar a integração entre aplicações e empresas, por exemplo, quando uma empresa adquire outra empresa, que possui uma infraestrutura tecnológica totalmente diferente, ela precisa integrar o novo sistema ao seu.

Porém, essa tarefa é complexa, pode levar muito tempo ou até mesmo comprometer a operacionalização de alguns dos seus setores. Uma solução para evitar esses problemas é a arquitetura orientada a serviços ou SOA (*Service Oriented Architectures*) (ERL, 2005).

SOA é um estilo arquitetural capaz de apoiar o paradigma de orientação a serviços. Para Mark (2006), os serviços podem ser interpretados por três tipos de visões: visão conceitual; visão de processo de negócio; e visão tecnológica. A primeira visão (conceitual) define um serviço por meio de suas características, escopo lógico e o valor que agrega ao negócio. A visão de processo de negócio define serviços como uma sequência de atividades de um processo. Na visão tecnológica, serviço é interpretado como uma abstração da solução tecnológica, ou seja, uma interface para uma implementação específica (aplicações comerciais, sistemas legados ou plataformas de integração).

O estilo SOA é baseado em padrões para a representação de dados, a exemplo do XML (*Extensible Markup Language*), que é uma linguagem de marcação que define a forma de exibição de um determinado conteúdo de um documento, utilizando *tags* (marcador), ou seja, como o conteúdo marcado será visualizado na tela (PAPAZOGLU *et al.*, 2008). Além disso, SOA utiliza o conceito de serviços ou WS (*web services*) para permitir interoperabilidade e comunicação entre aplicações e serviços (ERL, 2005). No entanto, SOA não está limitada a um ambiente de computação distribuída, apenas pelo uso de WS, pois também é preciso seguir os princípios de orientação a serviços (FUGITA, 2009).

Outra questão importante da SOA é o conceito de separação de assuntos, que consiste em dividir um problema grande em partes menores, para que a solução também possa ser decomposta em várias partes, onde cada parte deve tratar de um assunto específico que compõe a solução completa (ERL, 2005).

De acordo com Papazoglou *et al.* (2008), a estrutura básica da arquitetura SOA, que representa seu modelo funcional, é formada por três agentes de *software*, e suas interações, conforme mostra a Figura 3:

- Provedor de Serviço (*Service Provider*): implementa e disponibiliza os serviços, publicando suas descrições (registro de serviços) para que possam ser descobertos e consumidos pelas aplicações;
- Registro de Serviços (*Service Broker*): armazena e disponibiliza as descrições dos serviços

Figura 3 – Estrutura básica da SOA

Fonte: Papazoglou *et al.* (2008)

publicados;

- Consumidor de Serviço (*Service Consumer*): são as aplicações consumidoras dos serviços que consultam as descrições dos serviços (Registro de Serviços) e, com as informações em mãos, invocam os serviços do Provedor de Serviços.

Devido a importância da SOA no contexto corporativo, a organização para o avanço dos padrões de informação estruturada, chamada OASIS (*Organization for the Advancement of Structured Information Standards*), que representa um consórcio internacional para a especificação e evolução dos padrões de interoperabilidade, definiu um modelo de referência OASIS (2006) (MACKENZIE *et al.*, 2006) e uma arquitetura de referência (OASIS, 2008) como padrão para prover um entendimento comum da SOA e apoiar o desenvolvimento de sistemas orientados a serviços, propondo as seguintes especificações para todos os serviços relacionados direta ou indiretamente a SOA:

- Descrição de Serviço: fornece descrições padronizadas dos serviços para permitir sua descoberta e uso por outras aplicações;
- Visibilidade: os serviços devem estar sempre visíveis para os consumidores, fornecendo suas descrições e políticas de acesso;
- Interação: é a comunicação entre provedores e consumidores de serviços, por meio da troca de mensagens na rede;
- Políticas de serviço: são informações que descrevem as condições de uso, restrições e obrigações estabelecidas para o uso dos serviços;
- Contrato de serviço: é o acordo que deve ser firmado entre as partes envolvidas com o serviço (provedor e consumidor), concordando com as políticas de serviço;

- Contexto de execução: representa o caminho formado pelo conjunto de elementos (técnicos e de negócio) envolvidos na interação de serviços, entre provedor e consumidor;e
- Modelo de dados: refere-se ao modelo lógico de estruturação das informações associadas ao consumo do serviço.

Estas especificações foram definidas com base nos princípios que regem o paradigma da arquitetura SOA, que são (ERL, 2005):

- Reúso: o serviço pode ser reutilizado por outras aplicações e serviços;
- Granularidade: representa o tamanho, escala e nível de detalhamento do serviço;
- Baixo acoplamento: dependência mínima entre os elementos de um serviço ou dos próprios serviços entre si;
- Governança: representa os processos de alto nível que governam a SOA;
- Abstração: representação da lógica interna de um serviço por meio de sua interface (serviço é totalmente abstraído da sua implementação);
- Composição: capacidade de se compor com outro serviço, formando serviços compostos;
- Independência: são autônomos e independentes de plataforma;
- Interoperabilidade: interação com outros serviços, independentemente da plataforma e tecnologia de desenvolvimento, utilizando padrões abertos; e
- Descoberta: os serviços podem ser localizados por outros serviços.

Um serviço representa a unidade fundamental da arquitetura SOA e pode ser definido como uma funcionalidade específica de uma aplicação ou um módulo de negócio que encapsula componentes de *software*, e devem ser bem definidos, auto-contidos e independentes de estado, sendo expostos por interfaces que podem ser consumidas por outras aplicações (MARK, 2006). Além disso, os serviços podem ser compostos por outros serviços para oferecer funcionalidades mais complexas e abrangentes, fornecendo novos serviços compostos, por meio de coreografia ou orquestração de serviços (ERL, 2005).

Coreografia é um tipo descentralizado de composição de serviços, em que cada serviço participante sabe as ações que deve executar para criar o serviço composto (POSLAD, 2009). Para isso, é definida a sequência de mensagens que devem ser enviadas por cada serviço envolvido, podendo ser múltiplos serviços clientes e provedores que irão interagir de forma independente e descentralizada (PAPAZOGLU *et al.*, 2008). Esse tipo de composição permite maior escalabilidade, em relação à inclusão de novos componentes e equipamentos no

sistema, uma vez que é descentralizado e executado de maneira colaborativa entre os serviços (PAPAZOGLU *et al.*, 2008).

Por outro lado, a orquestração de serviços é uma forma de composição centralizada, regida por um coordenador central responsável por orquestrar as interações entre os serviços participantes. A orquestração utiliza um processo de negócio executável que inclui a lógica de negócio e é controlado por uma das partes envolvidas, para enviar mensagens aos serviços, baseado na ordem de execução das tarefas de cada um. Segundo Poslad (2009), a orquestração é a técnica de composição de serviços mais adotada pela comunidade científica e profissionais da área.

Para tratar das questões relacionadas à interoperabilidade e integração de serviços, principalmente os construídos em diferentes plataformas e linguagens de programação, SOA utiliza o conceito de barramento de serviços ou ESB (*Enterprise Service Bus*), que representa a infraestrutura utilizada para disponibilizar os serviços de diferentes sistemas que estão remotamente distribuídos em uma rede de TI (Tecnologia da Informação), servindo como mediador de serviços que reúne um conjunto de componentes e aplicações para formar serviços compostos (PAPAZOGLU *et al.*, 2008).

O ESB é um barramento de mensagens padronizadas, baseado em padrões abertos para permitir o desenvolvimento, implantação e gerenciamento de sistemas orientados a serviços. Mais especificamente, podemos entender o ESB como o canal de comunicação entre aplicações clientes e os serviços do provedor, responsável pelas questões de conectividade, controle dos registros e publicações dos serviços, roteamento das mensagens, conversão de dados e protocolos, orquestração de serviços, confiabilidade, segurança, etc.. O barramento utiliza um protocolo ou tecnologia de padrão aberto para uniformizar a troca de mensagens entre consumidor e provedor de serviços, tais como: **SOAP** (*Simple Object Access Protocol*) e **REST** (*Representational State Transfer*),

SOAP é um protocolo de comunicação bastante utilizado pelos serviços baseados em XML, que permite a interação de serviços independente de plataforma, tecnologias de desenvolvimento (linguagem de programação) e protocolo de transporte, podendo utilizar HTTP (*HyperText Transfer Protocol*), FTP (*File Transfer Protocol*), SMTP (*Simple Mail Transfer Protocol*), entre outros. Além do SOAP, outras terminologias também são adotadas na SOA, tais como **XML** (*eXtensible Markup Language*), **WSDL** (*Web Service Description Language*) e **UDDI** (*Universal, Discovery, Description and Integration*).

Para que os serviços possam ser encontrados e utilizados por outras aplicações, eles devem ser publicados em um repositório de serviços e fornecer descrições padronizadas com as informações necessárias para utilizá-los corretamente. WSDL e UDDI são os padrões utilizados na SOA para tratar dessas questões. A linguagem de descrição de serviços *Web* (WSDL) é um padrão utilizado para descrever *Web Services*, com as informações sobre os tipos de dados utilizados pelo serviço, operações que executada, formato das mensagens enviadas e recebidas,

e outras informações importantes para requisitar os serviços, sendo um documento escrito em XML. Já o protocolo **UDDI** é responsável por permitir que os serviços sejam descobertos em um registro público, por meio das descrições com as especificações dos tipos de valores, interfaces e demais parâmetros dos serviços publicados num repositório *web*, permitindo assim a interação indireta entre provedor e consumidor do serviço ([ERL, 2005](#)).

Por outro lado, REST é um estilo arquitetural derivado de outros estilos de arquiteturas, que utilizam serviços *Web* baseados no protocolo de comunicação HTTP, para permitir a interoperabilidade entre sistemas e serviços heterogêneos ([FIELDING, 2000](#)). REST é baseado em um conjunto de princípios e define uma série de restrições de arquiteturas de rede para facilitar o acesso aos recursos do sistema.

Os princípios da arquitetura de serviços REST tratam dos seguintes aspectos ([HANSEN, 2007](#)):

- Serviços são auto-contidos: cada requisição deve conter todas as informações necessárias para ser atendida pelo provedor de serviços;
- Serviços devem ter interface uniforme: serviços cliente e serviços do provedor devem enxergar a mesma interface (padrão);
- Os serviços são identificados por URI individuais;
- Os serviços só podem ser manipulados por quatro operações básicas utilizando os métodos: GET, POST, PUT e DELETE:
 1. GET: Recupera a informação de um recurso;
 2. POST: Cria uma nova entidade ou recurso;
 3. PUT: Atualiza os dados de um recurso informado;
 4. DELETE: Exclui os dados de um recurso informado.

O desenvolvimento de ambientes orientados a serviço pode oferecer diversos benefícios, entre eles maior flexibilidade para mudanças futuras, facilidade de manutenção e evolução do sistema, por meio da integração de novos componentes e serviços ou reuso de serviços já existentes, independente da linguagem de programação que este foi desenvolvido, entre outros ([PAPAZOGLU et al., 2008](#); [ERL, 2005](#)). Em geral, os benefícios valem tanto para serviços baseados em SOAP ou REST.

Desta forma, vale ressaltar que fica a cargo do desenvolvedor a escolha da tecnologia específica (SOAP ou REST) para o desenvolvimento do ambiente, a partir da instanciação da arquitetura proposta neste trabalho (MOA-RA).

2.3 Computação Ubíqua

Com os avanços tecnológicos, os processadores estão ficando cada vez menores, mais rápidos e incorporados na maioria dos atuais dispositivos digitais, assim como em alguns objetos comuns do nosso cotidiano como carros, televisão digital (*smart tv*), relógio (*smartwatch*), aparelhos de ar-condicionado, entre outros. Segundo [Augustin, Ferreira e Yamin \(2008\)](#), a tendência para o futuro próximo é que todas essas tecnologias estejam espalhadas ao nosso redor e conectadas entre si, por meio de sensores e redes sem fio, interagindo proativamente para auxiliar na execução das nossas tarefas do dia-a-dia, fornecendo serviços, recursos e informações para o usuário, em qualquer lugar e a qualquer hora. Este cenário, que descreve um ambiente tecnologicamente heterogêneo e conectado, ilustra características e fundamentos básicos do paradigma da computação ubíqua.

O termo “ubiquidade” significa estar presente em todo lugar ao mesmo tempo, e na computação ubíqua refere-se à integração das diferentes tecnologias imersas no ambiente do usuário, atuando de forma autônoma e quase imperceptível. Desta forma, a computação ubíqua é o paradigma computacional que investiga as questões relacionadas à ubiquidade tecnológica, abrangendo os princípios e tecnologias da computação móvel, computação pervasiva e redes de comunicação sem fio para prover serviços e informações em qualquer lugar e a qualquer momento, atuando de forma transparente ao usuário ([WEISER, 1999](#)).

A computação móvel trata das tecnologias dos diferentes dispositivos móveis com poder de processamento e comunicação por meio de redes sem fio, que são descobertas e conectadas dinamicamente pelo dispositivo, para permitir a comunicação com mobilidade. Dessa forma, o usuário tem acesso aos recursos computacionais, independentemente da sua localização ([FILHO, 2016](#)). Já a computação pervasiva significa que o “computador” está inserido no ambiente físico, imperceptível ao usuário, coletando informações desse ambiente para fornecer serviços adaptados ao contexto do usuário. O ambiente também pode detectar outros dispositivos que adentrem a ele, por meio de sensores. Dessa forma, [Zhao e Wang \(2011\)](#) deram um conceito mais abrangente definindo a computação pervasiva como tecnologias presentes no espaço físico, atuando proativamente e de forma transparente ao usuário. Essa característica representa o conceito da Internet das Coisas ou IoT (*Internet of Things*)¹, que podem ser ilustrado, por exemplo, por uma casa repleta de sensores e tecnologias que se conectam e interagem entre si para realizar tarefas adequadas a um contexto específico, tais como: ajustar a luminosidade e temperatura do ambiente, controlar o acesso, ligar dispositivos eletrônicos, etc.

Desta forma, a computação ubíqua representa a integração da computação móvel e

¹ É importante observar que a Internet das Coisas surgiu inicialmente como uma das partes tecnológicas rumo à implementação da ubiquidade. Porém, a computação ubíqua não necessariamente requer o uso de uma infraestrutura de Internet global ([MANWARING; CLARKE, 2015](#)). A definição do termo IoT ainda está em evolução, conforme discutido no trabalho de [Olson, Nolin e Nelhans \(2015\)](#), mas neste trabalho optou-se por utilizar o termo computação ubíqua.

pervasiva, aliadas às tecnologias de comunicação sem fio, atuando de forma transparente e proativa para fornecer serviços e informações apropriados ao usuário, levando em consideração o contexto atual e suas mudanças.

Segundo [Spínola e Travassos \(2012\)](#), as principais características identificadas nos sistemas ubíquos são:

- **Serviços onipresentes:** Os serviços podem ser utilizados a qualquer momento, independentemente da localização do usuário;
- **Invisibilidade:** Os serviços são entregues de forma espontânea e transparente ao usuário;
- **Sensibilidade ao contexto:** O sistema é capaz de identificar e coletar as informações do contexto que possam caracterizar uma entidade (objeto, pessoa, local) e ser utilizada para prover alguma ação em resposta;
- **Adaptação ao contexto:** O sistema altera o comportamento ou ajusta seus recursos para se adaptar ao contexto que foi identificado (*e.g.*, localização, perfil do usuário, configuração do dispositivo, etc.);
- **Captura de experiência:** O sistema coleta e registra as experiências do usuário, em relação às suas interações com o ambiente em que se encontra ou com o próprio sistema;
- **Descoberta de serviços:** Capacidade do sistema descobrir automaticamente os serviços disponíveis no seu raio de cobertura;
- **Interoperabilidade:** O sistema permite que o usuário possa utilizar, de forma transparente, os serviços e recursos computacionais de outros dispositivos do ambiente ubíquo;
- **Heterogeneidade tecnológica:** O sistema deve permitir sua utilização por dispositivos de plataformas de *hardware* e *software* heterogêneas;
- **Tolerância a falhas:** O sistema deve ser capaz de restaurar, após uma falha, ao seu estado anterior à falha, mantendo a integridade dos dados.

Vale ressaltar que, de acordo com as pesquisas de [Spínola e Travassos \(2012\)](#), não é necessário implementar todas estas características para que um sistema seja considerado ubíquo.

Uma característica importante dos ambientes ubíquos é a sensibilidade ao contexto (*Context-Sensitive*) ou ciência do contexto (*Context-Aware*) e está relacionada com a capacidade do sistema identificar, obter e gerenciar os elementos contextuais (dados e informações do contexto) para executar alguma tarefa específica ([MEDEIROS; TEDESCO; GOMES, 2011](#)). Sistemas com essa característica são chamados de sistemas sensíveis ao contexto ou CSS (*Context-Sensitive Systems*).

Desta forma, é importante saber o que é contexto e o que representa para os CSS. Para Spínola e Travassos (2012), contexto significa qualquer informação que possa caracterizar e interpretar a situação de uma entidade ou o estado dessa entidade em um contexto particular. Entidade pode ser uma pessoa, local ou objeto de interação do usuário. O contexto pode ser classificado de acordo com o tipo de informação que representa sua entidade, nas seguintes categorias: (1) Contexto computacional: informações sobre os recursos tecnológicos do usuário (dispositivo de acesso, configurações de *hardware* e *software*, rede de comunicação, status do GPS); (2) Contexto do usuário: informações pessoais, perfil, preferências, interações com o sistema, competências; (3) Contexto físico: localização geográfica, luminosidade do ambiente, temperatura; e (4) Contexto de tempo: hora, data, intervalo de tempo, histórico de acesso, entre outros (LOPES *et al.*, 2017).

2.4 Aprendizagem Móvel e Ubíqua

As tecnologias da computação ubíqua, assim como da computação móvel e pervasiva, possibilitaram novas oportunidades e benefícios para a educação, aplicadas como recursos didáticos de novos modelos de ensino e aprendizagem, como (i) Aprendizagem móvel ou *m-learning* (*mobile learning*) e (ii) Aprendizagem ubíqua ou *u-learning* (*ubiquitous learning*).

M-learning significa aprendizagem com mobilidade e consiste no uso de aparelhos celulares, *tablets* e PDA (*Personal Digital Assistant*), utilizando redes de comunicação sem fio para fins educacionais, proporcionando um modelo de aprendizagem com maior flexibilidade do local em que possa ocorrer (MARÇAL; ANDRADE; VIANA, 2019).

De acordo com Peng (2009), *m-learning* representa uma fase da evolução do *e-learning*, classificada cronologicamente em três fases. A Figura 4 mostra a evolução cronológica do *m-learning*.

Figura 4 – Fases do *M-learning*

Fonte: Filho (2016)

A primeira fase da *m-learning* teve origem com as primeiras iniciativas de utilização dos dispositivos móveis (PDA) em sala de aula, servindo apenas como um recurso didático novo e atraente para melhorar a motivação do aluno durante a aula.

Os avanços tecnológicos e a popularização dos dispositivos móveis (celular e *tablet*) com

acesso à internet e redes sem fio, deram origem a segunda fase da *m-learning*, que caracterizou de fato o modelo de aprendizagem móvel. Os dispositivos passaram a ser utilizados além das fronteiras físicas da escola, permitindo que a aprendizagem possa ocorrer em qualquer lugar e hora, com a mobilidade dos recursos didáticos e facilidade de acesso em locais informais de aprendizagem como jardins botânicos, zoológicos, centros históricos, museus, etc. (MARÇAL; ANDRADE; VIANA, 2019).

Por fim, a terceira fase representa a evolução do *m-learning* para *u-learning*, marcada pelo uso das tecnologias ubíqua em conjunto com os recursos da computação móvel para favorecer as práticas de ensino e aprendizagem, fornecendo novos serviços e vantagens para a educação. Neste paradigma, o ambiente é capaz de prover adaptações nos serviços e conteúdos didáticos às características e necessidades individuais do aluno, em qualquer lugar e no momento certo, baseado no contexto que representa uma determinada situação de aprendizagem (HWANG *et al.*, 2011).

Desta forma, os ambientes *u-learning* também são ambientes de aprendizagem sensíveis ao contexto, que utilizam dispositivos de captura (*e.g.* GPS e sensores) para monitorar e obter as informações necessárias para realizar as adaptações contínuas e aproveitar da melhor forma possível os recursos de aprendizagem, no contexto que está inserido (MACHADO *et al.*, 2017). Estas características podem trazer benefícios para o processo do ensino-aprendizagem, tais como: serviços personalizados; mobilidade para o estudo; materiais didáticos que se adaptam ao aluno; aprendizagem no momento certo (*just-in-time*); recomendação e entrega de conteúdos para o aluno; entre outros (MARCOLINO, 2019; FILHO, 2016).

Na educação, contexto pode ser qualquer informação que o ambiente de aprendizagem sensível ao contexto possa entender como uma situação de aprendizagem específica do aluno e utilizá-la a seu favor, como por exemplo: preferências de estudo; estilo de aprendizagem; atividades correntes; localização do aluno; dispositivo de acesso; entre outros (YU; ZHOU; SHU, 2010; MEDEIROS; TEDESCO; GOMES, 2011; OGATA; YANO, 2004) definiram os seguintes requisitos não-funcionais que os ambientes *u-learning* devem levar em consideração:

- Permanência: manter e proteger todos os trabalhos e registros das atividades realizadas pelos alunos;
- Acessibilidade: permitir o acesso aos materiais didáticos independentemente da localização do aluno;
- Imediatez: recuperar as informações requisitadas pelo aluno, de forma imediata;
- Interatividade: permitir a comunicação dos alunos entre si ou com professores, de forma síncrona ou assíncrona;
- Atividades instrucionais localizadas: apresentar as atividades de forma natural e considerando a situação de aprendizagem individual do aluno.

Os ambientes de aprendizagem ubíqua podem oferecer uma variedade de serviços e funcionalidades para apoiar as práticas de ensino e aprendizagem. Entre eles, destacam-se os seguintes serviços *u-learning*: (i) Recomendação de conteúdos: recomenda conteúdos de aprendizagem, de acordo com as preferências, estilo e necessidades do aluno; (ii) Apresentação: apresenta conteúdos e informações ao aluno, no momento apropriado; (iii) Entrega (*Delivery*): entrega materiais e recursos didáticos aos alunos adequados, de forma proativa; (iv) Adaptação: adapta conteúdos e serviços ao contexto do aluno; (v) Orientação/Navegação: utiliza as informações do contexto para conduzir o aluno na execução de uma atividade; e (vi) Autoria: facilita a construção de conteúdos e aplicações *u-learning* (LOPES *et al.*, 2017).

Em relação à construção de materiais didáticos, as ferramentas de autoria são programas utilizados para facilitar este processo, fornecendo mecanismos para que usuários sem conhecimento em programação possam criar conteúdos digitais e até mesmo aplicações *Web* ou aplicativos móveis, de maneira rápida e simples (FLÔRES; TARROUCO; REATEGUI, 2011).

Ferramentas de autoria podem contribuir para o processo de ensino e aprendizagem permitindo que professores de diferentes áreas do conhecimento construam seus próprios recursos didáticos, de diferentes tipos e formatos de mídia como objetos de aprendizagem (OA), exercícios interativos, *quizz*, microblog, animações, vídeos, entre outros. Além disso, também existem *softwares* de autoria voltados ao desenvolvimento de aplicações para dispositivos móveis, que permitem integrar tecnologias sensíveis ao contexto para gerar aplicativos *u-learning*.

Diante deste cenário, os ambientes de aprendizagem e autoria ubíqua podem oferecer vantagens para o processo educacional, por meio de facilidades, tecnologias, serviços e recursos potencialmente favoráveis à comunidade acadêmica de professores e alunos.

Por outro lado, o desenvolvimento de ambientes *u-learning* é uma tarefa complexa e demorada, devido a diversos fatores. A diversidade de informações do contexto que o ambiente educacional deve coletar e tratar corretamente, para fornecer serviços adaptados às necessidades de aprendizagem de cada aluno representa uma das dificuldades envolvidas neste processo (MACHADO *et al.*, 2017). Um quantidade de informações diferentes que o sistema deve analisar como variáveis independentes, ou mesmo associar com outras informações do contexto atual ou do passado, para interpretar a situação de aprendizagem correta sempre que houver uma mudança de contexto, é um processo difícil de ser implementado (PERNAS, 2012).

Além disso, construir sistemas para um ambiente tecnologicamente heterogêneo, que devem se adaptar às constantes mudanças de contexto, de forma eficiente e sem interromper seu funcionamento, é outro fator que aumenta a complexidade do desenvolvimento de ambientes *u-learning*. “A tarefa de adaptação é bastante complexa, uma vez que para determinar as alterações no contexto do ambiente muitas vezes não se avalia somente modificações isoladas, mas de vários aspectos combinados” (PERNAS, 2012).

Outra dificuldade está relacionada com o desenvolvimento de ferramentas de autoria com

geração automática do código da aplicação, baseado nas ferramentas da Engenharia Dirigida por Modelos ou MDE (*Model-Driven Engineering*), pois não são ferramentas intuitivas e fáceis de usar, além de não considerarem as condições do ambiente real, exigindo mais esforços dos engenheiros e programadores que forem utilizá-las (FILHO, 2016).

Por estes motivos, é importante e necessário o apoio arquitetural para ajudar na condução desse processo, fornecendo diretrizes e metodologias de reuso sistemático para o domínio em questão (PINHEIRO, 2019; DUARTE FILHO, 2016; FILHO, 2016; MACHADO *et al.*, 2017).

2.5 Considerações Finais

Neste capítulo foram apresentados os aspectos relevantes das áreas de embasamento teórico deste trabalho, relacionadas à arquitetura de *software*, com foco na arquitetura de referência e SOA, que representam os pilares desta proposta. Também foram apresentados os conceitos e princípios das áreas referentes ao domínio de aplicação da arquitetura MOA-RA, ligadas à computação ubíqua e aprendizagem móvel e ubíqua (domínio de aplicação).

No próximo capítulo apresenta-se as discussões sobre algumas arquiteturas de referência para ambientes *u-learning*, levantadas em mapeamentos sistemáticos da literatura realizados no âmbito deste trabalho.

ARQUITETURAS DE AMBIENTES DE APRENDIZAGEM MÓVEL E UBÍQUA

3.1 Considerações Iniciais

Este capítulo apresenta as discussões sobre alguns trabalhos relacionados à arquitetura de ambientes de aprendizagem ubíqua, que contribuíram direta ou indiretamente para o desenvolvimento desta tese de doutorado.

Os trabalhos serviram como fontes de informação para as etapas iniciais do desenvolvimento da arquitetura proposta, e foram obtidos por meio de um mapeamento sistemático da literatura (MSL), realizados no início deste trabalho, e complementado por outros trabalhos também relevantes, pesquisados nos repositórios digitais de tese e dissertação de algumas Universidades brasileiras (USP, UFRGS, UFC, UFMS), que não foram encontradas no MSL. As informações completas do MSL estão disponíveis no trabalho de [Lopes et al. \(2017\)](#).

A Seção 3.2 apresenta as descrições das arquiteturas do domínio *u-learning*. Na Seção 3.3 é apresentada a análise comparativa entre tais arquiteturas. Na Seção 3.4 são dadas as considerações finais do capítulo.

3.2 Arquiteturas *M-learning* e *U-learning*

No trabalho de [Moebert, Höfler e Lucke \(2016\)](#) é apresentado um sistema de autoria para representar as abstrações da aprendizagem móvel adaptativa em elementos com estrutura mais concreta, que possam ser manipulados mais facilmente. O sistema proposto faz parte de uma ampla gama de aplicações do projeto MOTIVATE (<http://motivate-project.de>), que oferece estrutura para o desenvolvimento de aplicativos de aprendizado móvel adaptativos.

A ferramenta é baseada em um conjunto de regras e relacionamentos utilizados para

representar as relações entre as unidades de aprendizagem, informações do contexto e definir as regras de adaptação.

Segundo os autores, nas aplicações adaptativas os motivos que ligam diferentes unidades de aprendizagem entre si, de forma implícita ou explícita, descrevem suas relações. Desta forma, eles definiram como relevantes, os seguintes tipos de relações (MOEBERT; HÖFLER; LUCKE, 2016):

- Pré-requisito (PRE): exigência necessária para uma unidade de aprendizagem;
- Base (BAS): requisito sugerido para uma unidade de aprendizagem;
- Alternativa (ALT): representação alternativa da mesma unidade de aprendizagem;
- Version (VER): versão mais antiga de uma unidade de aprendizagem;
- Referência (REF): referência a uma unidade de aprendizagem diferente;
- Extensão (EXT): conteúdo estendido para uma unidade de aprendizagem;
- Ajuda (HLP): conteúdo apresentável para uma unidade de aprendizagem.

As informações do contexto também são importantes para adaptar o conteúdo ao aluno. Portanto, é preciso encontrar uma forma que facilite a descrição das situações relevantes em determinadas unidades de aprendizagem. De acordo com a proposta do trabalho, isso pode ser feito adicionando informações específicas do contexto na unidade de aprendizagem. Tais informações serão usadas posteriormente para gerar regras de adaptação que ativam a unidade na situação apropriada.

Já as regras de adaptação são usadas para tornar o conteúdo educacional adaptado às características, preferências, necessidades e contexto do aluno. Elas são geradas com base nas informações do contexto (implícitas e explícitas) do cenário de aprendizagem, consideradas importante para a aplicação como local, horário, características do aluno e informações do conteúdo, por exemplo.

As unidades de aprendizagem são objetos de aprendizagem (OA), baseados no padrão LOM (*Learning Object Metadata*), e representam partes do conteúdo de um assunto e são ordenados em uma sequência de níveis (unidades), de ordem crescente. O relacionamento entre as unidades é feito de forma simplificada sem a necessidade de conhecimentos de lógica *Booleana*, apenas utilizando o método *Drag-and-Drop* (arraste e solte), e ocorre da seguinte forma: a unidade anterior é relacionada como “pré-requisito” para a unidade imediatamente posterior (Figura 5).

A outra relação que o ambiente de autoria permite é entre a unidade de aprendizagem e as informações do contexto, ou seja, cada unidade pode ser associada com diferentes informações do

Figura 5 – Relacionamento entre unidades de aprendizagem

Fonte: Moebert, Höfler e Lucke (2016)

contexto, de forma implícita ou explícita. Por exemplo, o usuário pode inserir suas informações pessoais (preferências, características, etc.) ou as informações podem ser obtidas dinamicamente por meio do dispositivo móvel do aluno, durante a execução de uma unidade de aprendizagem (localização, hora, temperatura).

A Figura 6 mostra uma unidade de aprendizagem (“*Learning Unit 2*”) com informações do contexto que lhes foram associadas pelo usuário do sistema. Ela ilustra a solução proposta no trabalho por meio da representação visual de regras de adaptação, tornando mais concretas algumas abstrações do modelo da aprendizagem móvel adaptativa.

Figura 6 – Associação entre unidade e informação do contexto

Fonte: Moebert, Höfler e Lucke (2016)

O trabalho de [Bravo et al. \(2016\)](#) apresenta a plataforma OPPIA (*OPPortunistic Intelligent Ambient learning*), que é um ambiente de aprendizagem colaborativa inteligente, que cria redes virtuais dinâmicas entre professores, alunos e especialistas, fornecendo atividades e recursos didáticos adaptados.

O ambiente utiliza tecnologias *Web-semântica*, técnicas de mineração de dados (*data mining*), serviços *context-aware* (contexto do usuário, físico e computacional) e recursos adaptativos e de personalização para prover mecanismos de compartilhamento dos recursos de *hardware* e *software* dos dispositivos conectados, criar dinamicamente redes e laboratórios virtuais para a troca de informações entre seus membros, gerar conteúdos baseados nas informações contextuais existentes (institucional e pessoal), entre outros.

A Figura 7 mostra a arquitetura da plataforma OPPIA, que é uma arquitetura de 5 camadas compostas por um conjunto de serviços em cada camada. A arquitetura está organizada da seguinte forma: (1) Camada de comunicação; (2) Camada de computação em nuvem; (3) Camada de gerenciamento do conhecimento; (4) Camada de sistema especialista e (5) Camada de serviços de aprendizagem.

Figura 7 – Camadas conceituais da plataforma OPPIA

Fonte: [Bravo et al. \(2016\)](#)

- Camada de comunicação: provê recursos para estabelecer automaticamente as conexões com os usuários. As redes esporádicas de aprendizagem colaborativa (professores, alunos e especialistas) são criadas sob redes *ad-hoc* já estabelecidas dinamicamente entre os dispositivos móveis próximos e a comunicação com os usuários remotos (local). Serviços das camadas superiores ocorre por meio de *links*, onde os protocolos e mecanismos para estabelecer estes *links* também são tratados nesta camada.
- Camada de computação em nuvem: responsável por fornecer recursos extras para os dispositivos móveis, por meio de serviços *web* de armazenamento, processamento e compartilhamento de informações. Além disso, esta camada também é responsável por criar laboratórios virtuais (na nuvem), considerando os usuários próximos, suas especialidades e os recursos disponíveis em seus dispositivos móveis, para fornecer serviços e acesso a *software* especialistas da sua área;

- Camada de Gerenciamento do Conhecimento: responsável pela gerência das informações pessoais e acadêmicas dos usuários para fornecer recursos mais adaptados às características individuais do aluno ou do grupo que faz parte. As informações são gerenciadas utilizando técnicas de mineração de dados, serviços de sistemas de recomendação e tecnologias da Web-semântica. A plataforma utilizou ontologias e *Dynamic Reusable Learning Objects* (DRLOs) para melhorar o gerenciamento e a modelagem dos conteúdos de aprendizagem.
- Camada de sistemas especialistas: trata dos aspectos de seleção, *design* de conteúdos, recursos e atividades de aprendizagem, por meio de um sistema especialista que incorpora um montador para construção de objetos de aprendizagem dinâmicos e reutilizáveis, a partir dos DRLOs presentes no repositório, utilizando três analisadores específicos (analisador de perfil, analisador de recurso e analisador de avaliação);
- Camada de serviços de aprendizagem: responsável pelo acesso aos conteúdos didáticos e serviços disponíveis na OPPIA. A camada possui três módulos para tratar destes aspectos: TICs inteligentes (aplicativos móveis, *desktop* e *web*), Sala de aula adaptável (salas virtuais adaptadas ao perfil dos usuários) e plano de aprendizagem inteligente (criados dinamicamente para uma determinada sessão de aprendizagem).

O trabalho de Escudero *et al.* (2018) apresenta uma ferramenta de autoria *m-learning* para o ensino e tratamento de pessoas com transtorno do espectro autista (TEA). A ferramenta foi proposta para auxiliar professores e profissionais da saúde a criar conteúdos e OA sobre TEA, para dispositivos móveis. Além disso, a ferramenta também serve para ajudar profissionais da saúde na produção de materiais para tratamentos personalizados e permite acompanhar a situação de cada paciente, em tempo real, por meio de *smartwatch* integrado no sistema.

O sistema consiste em uma aplicação *web* que se comunica com plataformas heterogêneas de *hardware* (*smartphone* e *smartwatch*) e *software* (*Android*), desenvolvido por meio de uma arquitetura cliente-servidor, baseada no padrão MVC e utilizando *web-semântica Restfull* (estilo Rest) para o ambiente ubíquo sensível ao contexto.

No trabalho de Hsu *et al.* (2015) foi proposto um ambiente de aprendizagem ubíqua baseado em tecnologia de identificação ativa por radiofrequência (RFID) para identificar a localização do aluno e fornecer apoio instantâneo durante as atividades de aula de campo, chamado ALESS (*Active Learning Support System*).

A Figura 8 mostra a arquitetura do sistema ALESS construída no modelo Cliente/Servidor e baseada no estilo SOA.

A arquitetura apresenta dois módulos e três bancos de dados, no lado Servidor. O módulo de configuração do ambiente acessa o banco de dados de parâmetros do ambiente para recuperar as informações do contexto e de configuração do sistema (*e.g.* localização, percursos, objetivos da aprendizagem, unidades de aprendizagem, metas do aluno, etc.).

Figura 8 – Arquitetura do ALESS

Fonte: Hsu *et al.* (2015)

O mecanismo de apoio à navegação adaptativa recebe a solicitação de orientação, enviada quando o usuário faz o *login* no sistema, e determina o melhor caminho da aprendizagem e armazena no banco de dados do caminho da aprendizagem. No banco de dados do aprendiz ficam armazenadas as todas as informações pessoais e educacionais de cada aluno cadastrado.

O lado Cliente também possui três bancos de dados e sete módulos responsáveis pelos seguintes serviços:

- Módulo de autenticação do usuário: verifica se os dados de login do usuário são válidos para acessar o sistema;
- Módulo de detecção da localização: determina a localização do usuário através da verificação das etiquetas RFID ativas (sensor RFID) e banco de dados de mapas; Módulo GUI do usuário: marca a localização do aluno no mapa que é fornecido a ele;
- Módulo de solicitação de orientação: é chamado quando detectar que o aluno está saindo do local corrente e indica o passo seguinte para o aluno;
- Módulo de exibição do mapa: mostra o mapa com a localização do aluno marcada;
- Módulo de exibição do material de aprendizagem: mostra o material didático automaticamente quando o aluno estiver próximo de uma meta específica da aprendizagem (etiqueta);
- Módulo de gravação do perfil: registra os movimentos de cada aluno durante o processo da aprendizagem no banco de dados do perfil de aprendizagem.

Em outro trabalho, [Duarte Filho \(2016\)](#) propõe uma arquitetura de referência orientada a serviços para o desenvolvimento de ambientes de aprendizagem móvel. A arquitetura, chamada “*Ref-mLearning*”, é do tipo cliente/servidor, composta por seis camadas que incorporam um conjunto de módulos de serviços para ambientes *m-learning*, baseados na SOA, sendo elas: Camada de Dados; Camada de Apresentação; Camada de Aplicação; Camada de Qualidade de Serviços; Camada de Visão e Camada da Intermediação de Serviços. A Figura 9 mostra a visão geral da *Ref-mLearning*.

Figura 9 – Visão geral da *Ref-mLearning*

Fonte: [Duarte Filho \(2016\)](#)

- Camada de Dados (*persistence layer*): armazena todos os dados criados e manipulados pelo ambiente, e os fornece quando requisitados. Está localizada no lado servidor e incorpora um módulo de persistência para garantir a integridade dos dados (SGBD):
- Camada de Apresentação (*presentation layer*): recebe as solicitações do cliente e realiza a apresentação visual das informações. Porém, antes da apresentação os dados originais são analisados para selecionar o estilo apropriado da informação que será apresentada, em seguida, a camada envia as informações estilizadas para o servidor da aplicação, que transfere para o cliente. A camada incorpora três módulos para tratar dos aspectos da SOA:
 - 1- Descritor de Serviços: define os tipos de dados trocados entre os serviços cliente e servidor (requisição – resposta);
 - 2- Controlador de Requisições: garante a sincronia dos serviços orquestrando a execução dos serviços dos outros módulos;
 - 3- Engine de Serviços: converte as mensagens das solicitações para o tipo de dados utilizado pelos serviços (aplicação);

- Camada de Qualidade de Serviços (QoS): monitora o cumprimento dos atributos de qualidade definidos nas camadas de serviços, registrando as inconformidades que forem encontradas. A camada está integrada com a Camada de Intermediação e parte da Camada de Apresentação;
- Camada da Intermediação de Serviços: controla e organiza os serviços do ambiente, permitindo que eles sejam descobertos, compostos e acessados adequadamente. Para isso, a camada incorpora os seguintes elementos:
 - 1- Registrador de serviços: responsável pelo registro dos serviços no repositório junto com os atributos de qualidade permitindo mapear os serviços disponíveis e monitorar os aspectos de qualidade;
 - 2- Agente de serviços (Mediador de serviços): relacionado com as questões de roteamento e transporte das mensagens trocadas entre os serviços cliente e servidor;
 - 3- Escalonador de serviços: verifica as dependências das linguagens descritivas das requisições, políticas de uso e requisitos de interoperabilidade definidos para os serviços solicitados.
- Camada de Aplicação (*application layer*): camada que reúne os serviços referentes às principais funcionalidades de um ambiente de aprendizagem móvel, incorporados entre seus oito módulos: (1) Módulo de Acesso; (2) Módulo de Administração; (3) Módulo de Documentação; (4) Módulo de Comunicação; (5) Módulo de Personalização; (6) Módulo de Adaptação ao Contexto; (7) Módulo de Ensino e (8) Módulo de Autoria:
 - (1) Módulo de Acesso: controla o acesso aos serviços do ambiente, verificando se o serviço requisitante pode ou não acessar o serviço solicitado;
 - (2) Módulo de Autoria: conforme o autor, este é um dos principais módulos da *Ref-mLearning*, responsável pelo apoio à elaboração dos conteúdos didáticos (materiais e avaliações). O módulo de autoria aborda questões sobre: (1) estruturação e modelagem de conteúdo (identificação e representações dos conceitos e seus relacionamentos); (2) edição de conteúdo (criação de mídias (vídeos, imagens, som) e documentos educacionais); (3) geração automática de conteúdo (construído em um formato específico, considerando a resolução da tela do dispositivo móvel); (4) compartilhamento e integração de conteúdo (uso de ontologias, dicionários de termos, glossários, entre outros); e (5) captura de conteúdo (captura e armazenamento de discussões e experiências educacionais dos usuários);
 - (3) Módulo de Personalização: é um módulo transversal (relacionado com os demais módulos da camada de aplicação). Seu objetivo é estabelecer mecanismos para criação e uso de modelos (*template*), opções de multilinguagem e adequação a padrões (tais como IMS, SCORM, LOM, entre outros). Suas funcionalidades afetam todos os demais, alterando o comportamento ao inserir funcionalidades de personalização;

- (4) Módulo de ensino: responsável pela apresentação e disponibilização dos conteúdos elaborados, bem como, é responsável pela avaliação do aluno, disponibilizando e aplicando exercícios, questionários, provas, entre outros. Outra questão atribuída ao módulo de ensino é a adaptabilidade dos conteúdos, por meio de parâmetros como conhecimento prévio (*background*), objetivos, interesses e perfil de aprendizagem individual do aluno, permitindo disponibilizar um mesmo conteúdo, de diferentes formas de estrutura e navegação;
- (5) Módulo de Colaboração/Comunicação: estabelece o tipo de comunicação utilizada pelo ambiente e permite aos usuários definir a forma de comunicação (SMS, MMS, voz ou teclado). Também reúne ferramentas síncronas e assíncronas (*chat*, *web conference*, *e-mail*, mensagens) e ferramentas colaborativas (*wikis*, grupos e fóruns);
- (6) Módulo de Adaptação ao Contexto: responsável pelos mecanismos de reconhecimento do contexto e captura das situações de aprendizagem do aluno no ambiente *m-learning*.

Em outro trabalho, Marcolino (2019) apresentou uma infraestrutura para apoiar o desenvolvimento de aplicações de aprendizagem móveis para o ensino de programação, direcionado para cursos de nível superior (graduação). A proposta visa auxiliar diretamente professores e programadores de aplicações *m-learning* no domínio em questão (programação), e, indiretamente, os usuários (alunos) das aplicações desenvolvidas nessa linha de produtos.

A infraestrutura proposta no trabalho consiste em uma ferramenta de autoria para *m-learning*, chamada TANGRAM (*an infrasTructure to support the development of mobile leArning applicatioNs for the teachinG and leaRning of progrAMming fundamentals*), um catálogo de requisitos para aplicações móveis (domínio de programação) e uma arquitetura da linha de produtos de *software* (LPS), baseada em serviços e micro-serviços.

A Figura 10 mostra a visão geral da arquitetura para aplicações móveis definida no trabalho.

A arquitetura permite a interoperabilidade dos serviços e microserviços do ambiente móvel permitindo que a comunicação possa ser síncrona ou assíncrona, por meio dos diferentes protocolos que podem ser utilizados durante a implementação da aplicação (i.e. HTTP/REST, protocolo de acesso a objeto simples (SOAP), fila de mensagens avançadas (AMQP) (*Advanced Message Queuing Protocol*) e transporte de fila de mensagens de telemetria (*message queue telemetry transport*)).

A arquitetura apresenta as camadas da estrutura das aplicações *m-learning*, além das camadas da infraestrutura e fontes externas, definidas da seguinte forma:

- Camada de apresentação: responsável por apresentar a interface gráfica para o usuário (*UI Components*) e interpretar suas interações na forma de comandos lógicos enviados para as

Figura 10 – Arquitetura das aplicações móveis

Fonte: (MARCOLINO, 2019)

outras camadas da aplicação (*Controller*);

- Camada de Negócios: incorpora as questões referentes aos processos de ensino/aprendizagem geral e específicos do domínio da programação para separar a apresentação visual dos dados da lógica de negócio da aplicação, fornecendo uma interface simplificada;
- Camada de Dados: responsável pelo acesso aos dados dos repositórios locais ou remotos, com apoio de serviços ortogonais/transversais da Camada de Serviços/Microserviços (*cross-cutting/orthogonal services layer*) que fornecem funções para o acesso e gerenciamento dos dados e configurações do ambiente;
- Camada de Serviços Ortogonais/Transversais: incorpora um conjunto de funcionalidades e *features* dos requisitos não funcionais, fornecidos como serviços transversais para os demais serviços da arquitetura. Nesta camada estão concentrados serviços de segurança, comunicação interna ou com fontes externas, mobilidade, configuração, etc.;
- Camada *API Gateway*: responsável por identificar os microserviços solicitados por outros serviços, redirecionando as requisições dos clientes (internos ou externos) e ajudando na gerência do tráfego de dados enviados e recebidos na aplicação;
- Camada de Repositórios (*Repositories Layer*): trata dos aspectos estruturais dos padrões utilizados por cada repositório de dados, que podem ficar hospedados em diferentes servidores *web*. Além disso, a camada também é responsável por coordenar o uso da

cache dos dispositivos e gerenciar os processos de persistência dos dados, disponibilizando interfaces no modelo DAO (*data Access object*)

O trabalho de [Tsiakmaki e Hartonas \(2013\)](#) apresenta a arquitetura CROP (Conceito, Recurso, Ordem, Produto), uma arquitetura de referência para objetos de aprendizagem (OA) adaptativos, baseada nas definições e conceitos estruturais, representando a ontologia OWL.

A arquitetura tem o foco na modelagem da estrutura dos OA adaptativos dos serviços de aprendizagem semântica no domínio educacional que são descritos por um conjunto de metadados, seguindo o padrão LOM. Além disso, cada objeto está devidamente associado a um designador que representa um estilo e características cognitivas da aprendizagem do aluno.

Outra característica da CROP é o suporte à composição de objetos, desde que atendam aos requisitos do aluno e requisitos para as modificações dinâmicas dos OA durante a aprendizagem, por meio da colaboração dos serviços de aprendizagem.

Cada Objeto de Aprendizagem tem um conceito de destino, ou seja, um termo que descreve o objetivo educacional do OA e alguns deles são pré-requisitos necessários para outros conceitos utilizarem o recurso. Todos os conceitos da arquitetura CROP fazem parte da ontologia de conteúdo, que serve para desvincular todos os objetivos educacionais relacionados ao objeto.

A aprendizagem é monitorada pelo processo “*Order*”, um processo que é habilitado para cada par Aluno-Objeto, para obter as informações necessárias para o serviço de aprendizagem construir a melhor relação possível entre estes pares.

As classes da Ontologia de conteúdo preenchem as instâncias da classe OA, na ontologia CROP, sendo elas as classes: *KConcept*, *KResource*, *KOrder* e *KProduct*.

1) ***KConcept***: As instâncias da classe *KConcept* são itens conceituais, ou seja, tudo que pode ser definido como um objetivo de aprendizagem/ensino.

2) ***KResource***: Um item do *KResource* é um recurso de aprendizagem, associado a um arquivo que contém o material de aprendizagem real. Cada item tem seu próprio LOM, que captura todas as informações de metadados relativas ao recurso (requisitos técnicos, informações de versão ou informações de autoria).

3) ***KProduct***: é a única classe do OA que difere do domínio da CROP, sendo utilizado pelas demais classes da ontologia como o componente (produto) que representa a instância de um determinado OA, ou seja, na CROP, *KProduct* e OA significam a mesma coisa.

4) ***KOrder***: É o gerente da interação entre o *KProduct* e o aprendiz. Processa qualquer informação disponível sobre o nível de conhecimento do aluno, estilo de aprendizagem e preferências, e constrói uma instância do *StudentModel*. Além disso, monitora a interação *KProduct* - Aluno, modifica adequadamente o *StudentModel* e fornece as informações necessárias para adaptar a execução do *KProduct* ao perfil do usuário.

A Figura 11 apresenta a estrutura geral do objeto de aprendizagem CROP (*CROP Object*), que é composta por 5 camadas que incorporam os serviços necessários para a adaptação dinâmica dos OA.

Figura 11 – Camadas do “*CROP Object*”

Fonte: Tsiakmaki e Hartonas (2013)

Todos os objetos CROP possuem um gráfico conceitual (*XConcept*) que é formado sempre que houver conexão entre instância de um objeto com um pré-requisito deste objeto. O mesmo ocorre nas camadas “*Execution Model*” (XModel) e “*Execution Graph*” (XGraph), que são duas camadas responsáveis pela obtenção do objeto executável da arquitetura.

A camada “*Execution Model*” fornece apoio para a implementação de estratégias instrucionais específicas, determinando a sequência de ações para o processo de aprendizagem diferente para os alunos.

A camada “*Execution Graph*” constrói seu gráfico (*XGraph*) a partir do gráfico “KRC” (*Knowledge Requirements Chart*) respeitando as relações já existentes de todos os demais OA com um determinado KRC. O XGraph também possibilita ao aluno incluir novos elementos (nó) para manipular sua sequência de ações da aprendizagem como “Diálogo” (conversas estáticas) e “Controle” (hora, local, média, etc.).

A camada “KRC” cria seu gráfico (KRC) em cima da camada “*Concept Graph*”, associando os OA de cada nó do KRC com seu “Conceito de destino” e respectivos pré-requisitos. O KRC também contém um nó ligado à localização física do material didático concreto (imagem, documento, etc.).

Em outro trabalho Zhao e Wang (2011) é proposta uma arquitetura orientada a serviços para entrega de aprendizagem ubíqua sensível ao contexto, com objetivo de fornecer serviços de aprendizagem adaptados ao contexto do aluno. Para isto, a arquitetura incorpora três serviços que utilizam bases de regras de aprendizagem para tratar possíveis conflitos de interesses que possam surgir entre eles (Serviço de adaptação, Serviço de transformação e Serviço de apresentação). A Figura 12 mostra a arquitetura proposta.

Figura 12 – Arquitetura SOA para aprendizagem sensível ao contexto

Fonte: Zhao e Wang (2011)

O serviço de adaptação é responsável pela recomendação de serviços *u-learning*, baseado nas informações e serviços de aprendizagem, tais como informações do perfil do aluno, situação e contexto da aprendizagem, serviços e conteúdos didáticos disponíveis no sistema. Em outras palavras, recomenda serviços e recursos mais adequados para a situação de aprendizagem atual.

O serviço de transformação é a interface entre o serviço de aprendizagem e a situação *u-learning*, dando o suporte para as mudanças dinâmicas do contexto da aprendizagem. É responsável por transformar um serviço original em serviço adaptado, reconstruindo informações e conteúdos ou ré-codificando serviços com base nas sugestões recebidas, para adaptá-los ao contexto e situação de aprendizagem.

O serviço de apresentação é responsável pela apresentação da mídia adaptada pelo sistema que será entregue ao aluno.

A arquitetura permite o uso de diferentes tecnologias e tipos de contexto, por meio do “módulo de reconhecimento de contexto” que gerencia informações enviadas por sensores, câmeras e GPS.

O trabalho de García *et al.* (2014) apresenta uma ferramenta de autoria de aplicações ubíquas para o ambiente tecnologicamente heterogêneo da IoT (*Internet of Things*). A ferramenta, chamada Midga, fornece uma interface gráfica para facilitar a construção de objetos que possam se conectar com as diferentes tecnologias imersas no ambiente do mundo real, gerando modelos do sistema em uma linguagem específica de domínio ou DSL (*Domain-Specific Language*).

Desta forma, a ferramenta permite criar objetos para o domínio de aplicação que o autor deseja utilizar, com o suporte das tecnologias IoT, sem precisar ter conhecimentos em programação (GARCÍA *et al.*, 2014).

A Figura 13 mostra a arquitetura da ferramenta de autoria Midga, que é composta por quatro camadas.

Figura 13 – Visão geral do Midga

Fonte: García *et al.* (2014)

A camada *Process Definition* é responsável pelos mecanismos de definição do processo, ou seja, fornece a interface de edição da ferramenta para que o usuário possa projetar sua aplicação, gerando uma especificação em XML, baseada na DSL do Midgar. O arquivo XML gerado pela ferramenta de edição é enviado para a camada *Service Generation*, que o processa para gerar o código-fonte de uma aplicação Java. Na camada *Processor and Object Manager*, o código Java é compilado para *Android* e a aplicação é disponibilizada no servidor. A camada *Objects* é responsável por manter o registro dos serviços disponíveis para a interação com o ambiente.

Esta arquitetura caracteriza o Midga como um sistema de propósito geral, para projetos de aplicações que executam em dispositivos móveis que podem integrar diferentes objetos IoT, focados naqueles que podem executar e capturar do contexto. Porém, a ferramenta não

oferece mecanismos para implementar funcionalidades importantes para o processo de ensino e aprendizagem de ambientes *u-learning*, como por exemplo: inserir recursos didáticos multimídia; adaptação dos recursos de aprendizagem; e interação entre usuários.

O UFC-Inventor (*Ubiquitous Field Classes Inventor*), proposto por Filho (2016), é um ambiente de autoria de aplicações *u-learning* para aulas de campo. O ambiente é composto por dois sistemas integrados e uma linguagem de modelagem, utilizados para gerar aplicações ubíquas e multiplataforma, a partir do modelo gráfico da aula de campo criado pelo professor na ferramenta.

A arquitetura do UFC-Inventor é apresentada na Figura 14, no qual é possível observar os componentes representados na sua visão geral, que são: o ML4UL (linguagem de modelagem do ambiente); módulo UFC-GLM (modelagem) e UFC-Generator (gerador de código).

Figura 14 – Visão geral do ambiente UFC-Inventor

Fonte: Filho (2016)

O componente ML4UL (*Modeling Language for Ubiquitous learning*) representa uma linguagem de modelagem de domínio específico, que utiliza um modelo de especificação aberta e reutilizável, para representar as aulas de campo ubíquas. O módulo UFC-GLM é responsável pela modelagem gráfica das aulas de campo modeladas na linguagem ML4UL, com recursos de computação ubíqua. O módulo UFC-Generator gera o código de instalação da aplicação ubíqua para diferentes sistemas operacionais. Os dois módulos interagem por meio do formato especificado pela ML4UL. O módulo UFC-GLM representa a ferramenta de autoria, que fornece a interface de edição gráfica para modelar as atividades de aprendizagem, definir o fluxo de execução, inserir as tecnologias ubíquas (sensores, GPS, tipo de comunicação sem fio), e converter o modelo gráfico da aplicação ubíqua para o formato da especificação ML4UL.

3.3 Análise Comparativa

Nesta seção são apresentados os resultados da análise comparativa feita com as arquiteturas propostas nos trabalhos apresentados na seção anterior. Os trabalhos foram obtidos no MSL, selecionados com base nos critérios definidos no protocolo de busca, e complementados por outros trabalhos igualmente relevantes para esta tese de doutorado, encontrados nos repositórios digitais de teses e dissertações de Universidades brasileiras (USP, UFRGS, UFC, entre outras).

Todos os trabalhos contribuíram de alguma forma como fonte de informação para esta pesquisa, e alguns foram às principais referências desta tese, a exemplo dos trabalhos de [Filho \(2016\)](#), [Duarte Filho \(2016\)](#), [Moebert, Höfler e Lucke \(2016\)](#) e [Hsu et al. \(2015\)](#). As arquiteturas foram analisadas e comparadas, levando em consideração um conjunto de critérios importantes para o desenvolvimento da arquitetura proposta neste trabalho, sendo eles:

- Arquitetura de referência (AR): indica se a arquitetura proposta no trabalho é uma arquitetura de referência ou não (arquitetura concreta). Este critério é importante para identificar os trabalhos diretamente relacionados com a proposta desta pesquisa;
- SOA: indica se a arquitetura é orientada a serviços. Este item verifica se a arquitetura foi desenvolvida com base no estilo SOA, para fornecer os serviços *u-learning* dos ambientes educacionais;
- Domínio educacional: indica se é uma arquitetura para ambientes educacionais *e-learning*, *m-learning* ou *u-learning*. Identifica o domínio da arquitetura, em relação ao tipo (modelo) de ambiente de aprendizagem que a arquitetura pode desenvolver;
- Uso de sensores: indica se a arquitetura permite o desenvolvimento de ambientes de aprendizagem sensíveis ao contexto, por meio do uso de sensores de captura (*e.g.* GPS, sensor de presença, RFID);
- Contexto: refere-se aos tipos de contexto (informações) utilizados na arquitetura, tais como: usuário (preferências, conhecimento, perfil...), físico (localização, temperatura..) e computacional (processador, tamanho da tela, memória...);
- Autoria: indica o tipo de autoria que a arquitetura fornece apoio para o desenvolvimento, tais como: *m-learning* (conteúdo e aplicativo móvel) ou *u-learning* (conteúdo e aplicativo sensível ao contexto);
- Geração de código: indica se a arquitetura utiliza mecanismos de modelagem e geração do código, na ferramenta de autoria proposta por ela.

A partir da análise das arquiteturas abordadas neste capítulo, foi feito um comparativo entre elas utilizando critérios relacionados à proposta deste trabalho. Com base nos resultados

Tabela 1 – Comparativo das arquiteturas dos trabalhos relacionados

Arquiteturas Analisadas	Arquitetura Referência	SOA	Domínio	Uso de Sensores	Contexto	Autoria	Geração Código
TANGRAM	Não	Sim	<i>m-learning</i>	Não	-	<i>m-learning</i>	-
OPPIA	Não	Não	<i>u-learning</i>	Sim	Usuário, Local e Tec.	<i>u-learning</i>	-
<i>Ref-mLearning</i>	Sim	Sim	<i>m-learning</i>	Não	Usuário	<i>m-learning</i>	Sim
UFC-Inventor	Não	Não	<i>u-learning</i>	Sim	Local	<i>u-learning</i>	Sim
ALESS	Não	Sim	<i>u-learning</i>	Sim	Usuário e Local	-	-
CROP	Sim	Não	<i>m-learning</i>	Não	Usuário	-	-
Midga	Não	Não	<i>u-learning</i>	Sim	Local	<i>u-learning</i>	Sim
Moebert et al. (2016)	Não	Não	<i>u-learning</i>	Sim	Usuário e Local	<i>u-learning</i>	Sim
Zhao e Jin (2011)	Sim	Sim	<i>u-learning</i>	Sim	Usuário e Local	-	-
Escudero et al. (2018)	Não	Sim	<i>u-learning</i>	Sim	Usuário e Local	<i>m-learning</i>	-

Fonte: Elaborada pelo autor.

deste comparativo, apresentados na Tabela 1, foi possível observar algumas questões importantes, comentadas a seguir:

- Nenhuma arquitetura analisada atendeu todos os critérios utilizados no comparativo;
- Apenas três trabalhos apresentaram propostas de arquitetura de referência (*Ref-mLearning*, CROP e a arquitetura do trabalho de [Zhao e Wang \(2011\)](#)). As demais são arquiteturas concretas dos ambientes educacionais propostos nos respectivos trabalhos;
- Cinco arquiteturas abordam o conceito de serviços SOA, nas quais apenas duas são arquiteturas de referência orientadas a serviços (*Ref-mLearning* e arquitetura de [Zhao e Wang \(2011\)](#));
- A maioria são arquiteturas para ambientes *u-learning*, que fazem uso de tecnologias ubíquas como sensores, GPS, RFID (ALESS), *Smartwatch* ([ESCUDERO et al., 2018](#));
- Quase todas as arquiteturas utilizam informações do contexto para tratar de algum tipo de serviço como recomendação, adaptação, orientação, apresentação, entre outros, exceto a TANGRAM. Basicamente, as arquiteturas analisadas utilizam apenas dois grupos de informações do contexto, sendo: Contexto do usuário (perfil, conhecimento, preferências do aluno, interações com o ambiente, etc.) e Contexto da localização (latitude/longitude e IP da rede). Apenas a arquitetura OPPIA utiliza também as informações do contexto tecnológico ou computacional, referente às configurações de *hardware* do dispositivo do usuário (e.g. processador, memória, banda da rede);

- Apenas três arquiteturas fornecem apoio para o desenvolvimento de ambientes de autoria que permitem ao usuário definir as tecnologias e serviços *u-learning* que serão incorporados na aplicação que irá construir: *UFC-Inventor*; *Midga*; e a arquitetura proposta por [Moebert, Höfler e Lucke \(2016\)](#). Todas elas empregam ferramentas de modelagem e geração automática da aplicação, assim como a *Ref-mLearning*, só que para criar conteúdos *m-learning*. As demais arquiteturas relacionadas a este serviço são para autoria de objetos de aprendizagem (OA), autoria de conteúdos de aprendizagem móvel ou ubíqua, sem necessariamente precisar gerar o código da aplicação.

3.4 Considerações Finais

Neste capítulo foram apresentadas as arquiteturas consideradas mais relevantes no domínio dos ambientes de aprendizagem ubíqua e móvel. Vale ressaltar que outras arquiteturas também importantes no contexto deste trabalho são discutidas apenas no capítulo seguinte (Capítulo 4) para não ficar muito redundante com este capítulo, que ainda apresentou uma análise comparativa das arquiteturas, baseado num conjunto de critérios comuns e importantes para este trabalho.

A partir deste comparativo, foi possível identificar as diferentes características e soluções propostas em cada arquitetura, e observar que entre elas, as únicas com propostas relacionadas à autoria de aplicações *u-learning* que oferece mecanismos para o usuário inserir tecnologias sensíveis ao contexto nas funcionalidades da aplicação (*UFC-Inventor* e *Midga*), não são arquiteturas de referência para apoiar o desenvolvimento de novos ambientes de autoria com características similares, o que pode indicar uma lacuna na área.

O próximo capítulo discute sobre o processo de desenvolvimento da arquitetura de referência MOA-RA, proposta no presente trabalho.

DESENVOLVIMENTO DA ARQUITETURA DE REFERÊNCIA PROPOSTA

4.1 Considerações iniciais

Este capítulo apresenta o processo de desenvolvimento da arquitetura de referência (RA) proposta nesta pesquisa de doutorado. Com base nas investigações realizadas e considerando as vantagens da SOA, principalmente em relação às questões referentes à interoperabilidade entre aplicações heterogêneas, reúso dos serviços e separação de assuntos, que consiste em separar o problema em partes menores, foi definida uma arquitetura de referência orientada a serviços com a proposta de contribuir para projetos arquiteturais e desenvolvimento de ambientes de aprendizagem ubíqua, com foco no serviço de autoria de aplicações sensíveis ao contexto.

A arquitetura foi construída seguindo o processo ProSA-RA ([NAKAGAWA, 2006](#)), que é um modelo de referência para a construção e evolução de arquiteturas de referência, composto por quatro etapas específicas: (RA-1) Investigação das Fontes de Informação; (RA-2) Estabelecimento dos Requisitos Arquiteturais; (RA-3) Projeto da Arquitetura de Referência; e (RA-4) Avaliação da Arquitetura de Referência. O ProSA-RA foi adotado por ser um processo simples, consolidado, de fácil aplicação e que pode ser utilizado na construção de arquiteturas de referência em diferentes domínios de aplicação.

O restante do capítulo está organizado da seguinte forma. A Seção [4.2](#) corresponde à etapa de investigação das fontes de informação (RA-1), apresentando o resultado da análise dos trabalhos selecionados nos mapeamentos e das fontes encontradas nos bancos de teses e dissertações de importantes Universidades brasileiras. A Seção [4.3](#) trata sobre os requisitos da arquitetura de referência (RA-2) levantados a partir da investigação das fontes e definidos com base nas características e objetivos traçados para a arquitetura proposta. Na Seção [4.4](#) são discutidas as questões relacionadas ao estabelecimento da AR (etapa RA-3), apresentando os elementos que representam e descrevem os aspectos da arquitetura, em termos de modelagem,

relacionamentos, infraestrutura, módulos, serviços, entre outros. A Seção 4.5 apresenta uma breve descrição sobre os métodos utilizados na avaliação da arquitetura de referência, executados na etapa RA-4, sem entrar em detalhes, uma vez que as discussões sobre a avaliação arquitetural realizada neste trabalho são apresentadas integralmente no Capítulo 5. Por fim, na Seção 4.6 são apresentadas as considerações finais do capítulo.

4.2 Investigação das fontes de informação (RA-1)

A primeira etapa do processo de desenvolvimento (RA-1) é a etapa de investigação das fontes de informação, que foi realizada neste trabalho por meio de dois mapeamentos sistemáticos da literatura (MSL), complementada por pesquisas bibliográficas. A etapa serviu para identificar o estado da arte e levantar as informações que ajudaram a definir as características gerais da arquitetura e os requisitos importantes para o domínio de aplicação da arquitetura proposta.

Os MSL foram feitos durante as pesquisas iniciais desta tese, em momentos diferentes. O primeiro mapeamento identificou as fontes relacionadas aos ambientes de aprendizagem ubíqua, móvel e sensível ao contexto, e as informações completas deste MSL estão disponíveis em [Lopes et al. \(2017\)](#). Já o segundo mapeamento investigou as arquiteturas dos ambientes de autoria *u-learning* (móvel e sensível ao contexto) propostas na literatura, e seus resultados ainda não foram publicados.

As informações foram extraídas das fontes por meio da análise individual dos trabalhos, focada mais especificamente no levantamento de três grupos de informação distintas, sobre: (1) Ambientes *u-learning* ou ULE (*Ubiquitous Learning Environments*); (2) Ambientes educacionais genéricos ou GEE (*Generic Educational Environments*); e (3) Arquitetura orientada a serviços ou SOA (*Service-Oriented Architecture*). Esta decisão foi motivada por outros trabalhos que também adotaram o modelo ProSA-RA aplicada no domínio educacional, tais como [Duarte Filho \(2016\)](#), [Soad et al. \(2017\)](#), [Nakagawa, Sasaki e Maldonado \(2009\)](#), [Fioravanti, Nakagawa e Barbosa \(2010\)](#) e [Affonso e Nakagawa \(2013\)](#).

É importante destacar que alguns trabalhos investigados nesta etapa foram utilizados como fonte de informação de dois ou mais grupos, por abrangerem no seu escopo mais de uma temática que define cada grupo. Por exemplo, o trabalho de [Pinheiro \(2019\)](#) foi analisado em dois grupos (ULE e GEE) por abordar questões de natureza geral dos ambientes educacionais, assim como específicas da *u-learning*. Já [Duarte Filho \(2016\)](#), que apresenta uma proposta mais abrangente, baseada em arquitetura SOA para ambientes educacionais genéricos e focado na aprendizagem móvel, foi utilizado como fonte dos três grupos de informação (ULE, GEE e SOA).

A seguir, são discutidos os resultados da investigação, apresentando uma síntese da análise das fontes de cada grupo e comparando as informações obtidas. Para mais informações sobre a análise dos trabalhos, consulte o Capítulo 3.

4.2.1 Investigação do Grupo dos Ambientes de Aprendizagem Ubíqua

O grupo ULE (*Ubiquitous Learning Environment*) considera tanto as informações relacionadas ao domínio da aprendizagem ubíqua como relacionadas aos domínios da aprendizagem móvel e sensibilidade ao contexto, por serem áreas que fazem parte do escopo da computação ubíqua. Desta forma, a investigação das informações deste grupo analisou trabalhos com proposta diversificada, em relação ao domínio e foco principal da pesquisa, como por exemplo: o trabalho de [Filho \(2016\)](#) propôs um ambiente de autoria ubíqua; [Duarte Filho \(2016\)](#) apresentou uma arquitetura de referência para ambientes *m-learning*; o trabalho de [Bravo et al. \(2016\)](#) propôs um ambiente de aprendizagem sensível ao contexto; entre outros.

No trabalho de [Bravo et al. \(2016\)](#), foi utilizada uma arquitetura baseada em serviços para desenvolver um ambiente de aprendizagem colaborativa, que utiliza diferentes informações do contexto para criar redes virtuais de aprendizagem e recursos didáticos adaptados aos usuários. A arquitetura é composta por um conjunto de serviços *web* organizados em cinco camadas, para prover as principais funcionalidades e características do ambiente proposto, tais como as questões relacionadas à sensibilidade ao contexto, mobilidade, portabilidade, adaptação e funções de autoria, recomendação e entrega de objetos de aprendizagem adaptados ao contexto.

A arquitetura também incorpora técnicas de mineração de dados (*data mining*), *web* semântica e um sistema especialista para tratar da modelagem e gerenciamento dos objetos de aprendizagem adaptativos.

Em outro trabalho, [Moebert, Höfler e Lucke \(2016\)](#) propõem uma ferramenta de autoria de OA adaptativos para dispositivos móveis. A ferramenta utiliza regras de relacionamento para modelar graficamente as abstrações de adaptação que podem ser inseridas na relação entre unidades de aprendizagem e informações do contexto, na forma de elementos que possam ser facilmente manipuláveis pelo usuário (professor).

Para alcançar este objetivo o trabalho abordou diferentes conceitos que serviram para fundamentar sua proposta, tais como lógica *booleana*, mobilidade, autoria de objetos de aprendizagem móveis, sensibilidade e adaptação ao contexto.

O trabalho de [Duarte Filho \(2016\)](#) propõe uma arquitetura de referência para ambientes de aprendizagem móvel, chamada *Ref-mLearning*. A arquitetura é baseada no padrão MVC e estilo SOA para oferecer um conjunto de serviços *web* relacionados aos diferentes aspectos e funcionalidades propostas na arquitetura. Em relação específica ao domínio educacional, a *Ref-mLearning* incorpora tanto serviços da aprendizagem móvel e sensível ao contexto como serviços educacionais genéricos.

A análise da arquitetura *Ref-mLearning* permitiu identificar um conjunto de características e funcionalidades dos ambientes relacionados ao domínio da aprendizagem ubíqua/móvel (grupo ULE), mapeados a partir dos requisitos definidos na arquitetura. Tais características são

relacionadas aos seguintes conceitos:

- Adaptação ao contexto: a arquitetura possui um módulo (“Módulo de Adaptação ao Contexto”) responsável pelos mecanismos de sensibilidade e captura do contexto para tratar de serviços de aprendizagem adaptado ao aluno;
- Mobilidade e portabilidade: a arquitetura trata destes aspectos atendendo um conjunto de requisitos relacionados às questões de mobilidade e portabilidade do ambiente definindo, que foram levados em consideração no desenvolvimento da arquitetura;
- Autoria *m-learning*: a arquitetura fornece apoio para a construção de conteúdos didáticos em diferentes mídias (texto, áudio, imagem), por meio de um conjunto de serviços abordados no “Módulo de Autoria” da arquitetura.

[Pinheiro \(2019\)](#) propôs um ambiente *u-learning*, chamado Edubi, que consiste em um conjunto de aplicações para auxiliar a interação dos usuários com conteúdos didáticos adaptados às suas preferências de estudo, servindo como uma extensão para ambientes educacionais baseados no padrão LMS (*learning management system*) que pode integrar as funcionalidades de redes sociais, serviços baseados na localização (geolocalização), ferramentas de aprendizagem colaborativa (*chat*, grupos de discussão e compartilhamento de recursos), autoria, recomendação e notificação de conteúdos, entre outros, como por exemplo, alguns serviços disponíveis na plataforma do *Moodle*, *Wikipedia* e *Youtube*.

A arquitetura do Edubi segue o modelo cliente/servidor, onde o lado cliente é formado por quatro aplicações clientes para permitir o acesso por múltiplas plataformas (*Web*, *Mobile*, *Watch* e *TV*) e no lado servidor possui uma camada de integração (Componente de integração) com dois componentes (“Recomendador de conteúdos” e “Integrador de LMS”) responsáveis pela composição de algumas funcionalidades do *Moodle*, *Wikipedia* e *Youtube*, nas aplicações cliente do Edubi.

Estas funcionalidades são acessadas por meio do “Youubi API”, que fornece uma interface para a requisição dos serviços (*web-service* REST) disponíveis no “Youubi Server” ([MONTEIRO et al., 2015](#)), que é uma aplicação integrada no ambiente Edubi.

Em outro trabalho, [Marcolino \(2019\)](#) propõe uma ferramenta de autoria *m-learning* para auxiliar a aprendizagem de programação de software, chamada TANGRAM. A arquitetura do TANGRAM é do tipo cliente-servidor, baseada em serviços e microserviços *web*, que podem ser requisitados por meio de diferentes protocolos de comunicação, tais como: HTTP/REST, SOAP (*simple object Access protocol*) e AMQP (*Advanced Message Queuing Protocol*).

No lado cliente, a arquitetura da aplicação do usuário (aplicativo móvel) é formada pelas seguintes camadas

- Camada de apresentação: responsável pela interface gráfica do ambiente e conteúdos de aprendizagem apresentada no dispositivo móvel do usuário;
- Camada de Negócios: separa a lógica de negócio da apresentação visual dos dados para simplificar a interface da aplicação;
- Camada de Dados: responsável pelo gerenciamento dos dados armazenados nos bancos locais (memória do dispositivo) e remotos (repositório *web*);

No lado servidor estão concentrados os serviços (e microserviços) responsáveis pela maioria das funcionalidades (genéricas ou educacionais) oferecidas no ambiente e *features* dos requisitos não-funcionais fornecidos para os outros serviços tratados pela arquitetura. Estes serviços estão incorporados na “Camada de Serviços Ortogonais” e tratam das questões de segurança, mobilidade, configuração, comunicação, entre outros.

As requisições aos serviços e tráfego dos dados de envio e resposta entre aplicação cliente e servidor são gerenciadas pelo “*API Gateway*” que localiza os serviços, redireciona as requisições e envia o resultado para o usuário solicitante.

O trabalho de [Filho \(2016\)](#) apresenta o ambiente UFC-Inventor (*Ubiquitous Field Classes Inventor*), um ambiente de autoria de aplicações de aprendizagem ubíquas para aulas de campo. A proposta é facilitar a construção de aplicações *u-learning*, por parte de professores sem conhecimento em programação, com uma ferramenta (UFC-Inventor) que faça a modelagem e geração automática de aplicações multiplataforma, integrando tecnologias da computação ubíqua.

A arquitetura do ambiente é composta por três componentes básicos que atuam de forma conjunta na construção das aplicações *u-learning*, sendo: (i) UFC-GLM (módulo de modelagem gráfica); (ii) ML4UL (linguagem de modelagem para *u-learning*); e (iii) UFC-Generator (gerador da aplicação).

O módulo UFC-GLM é responsável pela modelagem gráfica das aulas de campo que permite ao usuário incorporar tecnologias ubíquas na aplicação que será construída. Para isso, o módulo disponibiliza um editor gráfico para o autor definir a atividade de aprendizagem, fluxo das ações da atividade, inserir os recursos da computação ubíqua que serão integrados na aplicação (*e.g.* sensores, GPS, informações e regras do contexto, ferramentas IMS, etc.). Ao final, o modelo gráfico gerado pelo UFC-GLM é convertido para uma especificação no formato da linguagem de modelagem ML4UL e enviado para o módulo UFC-Generator que é responsável por gerar o código-fonte da aplicação e disponibilizar para os alunos.

Neste trabalho, ainda são abordadas outras questões relacionadas ao ambiente proposto, tais como: portabilidade do ambiente para diferentes plataformas; mobilidade das aplicações geradas para dispositivos móveis; sensibilidade e adaptação ao contexto, por meio da capacidade das aplicações em identificar, capturar, armazenar e utilizar as informações do contexto da

aprendizagem para prover ações necessárias; e entrega das aplicações criadas para os devidos alunos.

Vale ressaltar que, além destas fontes, outros trabalhos da literatura também foram analisados nesta etapa e os resultados contribuíram para esta pesquisa, ajudando a identificar um conjunto de informações importantes para o domínio *u-learning*, que apoiaram o desenvolvimento da arquitetura proposta. Entre eles, estão os trabalhos de Escudero *et al.* (2018), Passos (2016), Castro *et al.* (2016), Hsu *et al.* (2015), Tsiakmaki e Hartonas (2013) e Zhao e Wang (2011).

A Tabela 2 mostra o comparativo dos trabalhos investigados, em relação aos conceitos-chave do domínio dos ambientes de aprendizagem ubíqua.

Tabela 2 – Conceitos do grupo ULE

Fontes	Sensível	Mobile	Autoria	Adapt.	Desc.	Recom.	Entrega
Pinheiro (2019)	X	X	X	X	-	X	-
Marcolino (2019)	-	X	X	-	-	-	-
Bravo <i>et al.</i> (2016)	X	X	X	X	-	X	X
Moebert, Hofer e Lucke (2016)	X	X	X	X	-	-	-
Duarte Filho (2016)	X	X	X	X	-	-	-
Filho (2016)	X	X	X	X	X	-	-X
Escudero <i>et al.</i> (2018)	X	X	X	-	-	-	-
Passos (2016)	X	X	-	X	-	X	X
Castro <i>et al.</i> (2016)	X	X	X	-	X	X	X
Hsu <i>et al.</i> (2015)	X	-	-	X	X	X	X
Zhao e Wang (2011)	X	X	-	X	X	X	-
Tsiakmaki e Hartonas (2013)	X	-	X	X	-	-	-

Fonte: Elaborada pelo autor.

Conforme a Tabela 2, pode-se observar que o conceito “Sensibilidade ao contexto” foi o conceito abordado pelo maior número de trabalhos (11 fontes), seguido por “Mobilidade/Portabilidade” e “Adaptação ao contexto” encontrados em 10 trabalhos no total. De fato, estes conceitos representam uma parte importante que caracteriza a essência da *u-learning* e servem como base de sustentação para os demais serviços deste domínio, sendo:

- **Sensibilidade ao contexto:** capacidade de identificar, obter e utilizar informações do contexto para prover determinado serviço (educacional), de forma proativa;
- **Mobilidade/Portabilidade:** relacionado à facilidade para o transporte e acesso aos recursos didáticos em diferentes lugares e dispositivos de *hardware* e *software*, mesmo durante a locomoção do usuário, permitindo a aprendizagem com mobilidade;
- **Autoria de conteúdo (*m-learning/u-learning*):** relacionado à construção de recursos didáticos e aplicativos de aprendizagem móvel sensíveis ao contexto;

- **Adaptação ao contexto:** capacidade do ambiente educacional (ou objetos de aprendizagem) se adaptar ao novo contexto do usuário, modificando o comportamento, configurações ou serviços para se ajustar ao contexto atual da aprendizagem;
- **Descoberta de recursos:** relacionado à descoberta automática de serviços, recursos didáticos e usuários próximos da localização atual do aluno;
- **Recomendação:** o sistema recomenda conteúdos e objetos de aprendizagem baseado nas preferências, localização ou perfil do aluno;
- **Entrega (*Delivery*):** o sistema entrega ao aluno conteúdos e materiais didáticos adequados ao seu contexto atual, de forma proativa.

4.2.2 Investigação do Grupo dos Ambientes Educacionais Genéricos (GEE)

Em relação ao grupo GEE, a investigação das fontes procurou identificar informações relacionadas às características e serviços educacionais genéricos presentes em diferentes tipos de ambientes de aprendizagem e seus domínios de aplicação, ou seja, conceitos-chave comuns entre ambientes *e-learning*, *u-learning* e *m-learning*.

Estes conceitos são relacionados a um conjunto de funcionalidades básicas para o apoio de práticas educativas e serviços administrativos dos ambientes educacionais, por exemplo, segurança do ambiente, disponibilização dos conteúdos de aprendizagem, gerência dos usuários, avaliação do aluno, entre outros serviços que possam ser usados por aplicações de plataformas heterogêneas (*Web*, *Mobile*, *desktop*, etc.).

Considerando estas características, as fontes investigadas neste grupo (GEE) foram os trabalhos com propostas focadas em ambientes educacionais genéricos ou arquiteturas de referência para o domínio educacional. Desta forma, foram selecionadas e analisadas as seguintes fontes de informação (GEE): [Pinheiro \(2019\)](#), [Duarte Filho \(2016\)](#), [Tsiakmaki e Hartonas \(2013\)](#), [Marcolino \(2019\)](#), [Zhao e Wang \(2011\)](#), [Zhao e Jin \(2011\)](#), [Fioravanti, Nakagawa e Barbosa \(2010\)](#), [Escudero *et al.* \(2018\)](#), [Silva \(2017\)](#), [Popolin e Fofonca \(2017\)](#), [Filho \(2016\)](#), [Hsu *et al.* \(2015\)](#), [Passos \(2016\)](#), [Castro *et al.* \(2016\)](#) e [Morales *et al.* \(2015\)](#).

O trabalho de [Fioravanti, Nakagawa e Barbosa \(2010\)](#) apresenta uma arquitetura de referência para sistemas *e-learning*, baseada no modelo cliente-servidor e formada por três camadas (Apresentação, Aplicação e Persistência). A camada de Aplicação concentra os principais serviços educacionais da arquitetura incorporados em um conjunto de módulos de serviços, devidamente relacionados entre si para tratar das questões referentes aos seguintes conceitos de domínio dos ambientes de aprendizagem genéricos (GEE): Documentação; Comunicação; Adaptação; Autoria; Avaliação; Administração; Adequação a padrões; Disponibilização; e Personalização.

Estes conceitos também foram incorporados na arquitetura de [Duarte Filho \(2016\)](#), com pequenos ajustes e adaptações para o domínio da aprendizagem móvel e orientação a serviços. Outros conceitos importantes para os ambientes educacionais, assim como para qualquer tipo de aplicação, estão relacionados a um conjunto de funcionalidades gerais de apoio aos sistemas, representados no grupo de serviços ortogonais da camada de Aplicação da arquitetura (*e.g.* segurança, persistência de dados, entre outros).

Em [Marcolino \(2019\)](#), foi estabelecido um catálogo de requisitos para ambientes de aprendizagem móvel, no domínio de programação. O catálogo define um total de 97 requisitos, classificados em 9 categorias: (1) Conteúdos de Aprendizagem; (2) Atividades de Aprendizagem; (3) *Feedback* e Resultados; (4) Monitoramento e Desempenho de Aprendizagem; (5) Mecanismos para Ensino de Programação; (6) Avaliação de Aprendizagem; (7) Apoio ao Ensino e Aprendizagem; (8) Usuários; e (9) Requisitos de Qualidade (não funcionais).

Analizando apenas os aspectos relacionados ao grupo GEE, foi possível identificar diferentes características e funcionalidades definidas nos requisitos de ambientes educacionais genéricos. Por exemplo, a arquitetura de referência incorpora requisitos para apoiar o desenvolvimento de aplicações educacionais que permita:

- **Conteúdos de Aprendizagem:** Gerenciar, disponibilizar, apresentar e criar conteúdos didáticos simples (texto, áudio, vídeo, etc.) ou interativos (exercício de múltipla escolha, pergunta e resposta, jogo educativo);
- **Atividades de Aprendizagem:** Gerenciar e executar diferentes tipos de atividades e tarefas de aprendizagem, enviar respostas, controlar o acesso e disponibilizar para *download*;
- **Feedback e Resultados:** fornecer respostas, correções e resultados das avaliações, baseado nas interações do usuário com os conteúdos disponíveis no ambiente;
- **Monitoramento e Desempenho de Aprendizagem:** monitorar e registrar o progresso e desempenho do aluno nas atividades, gerenciar, restaurar e gerar relatórios das sessões dos alunos nas atividades educacionais;
- **Avaliação de Aprendizagem:** fornecer um ambiente para execução de provas (*e.g.* questionários, pergunta e resposta, múltipla escolha) e avaliação automatizada ou semi-automatizada da aprendizagem do aluno;
- **Apoio ao Ensino e Aprendizagem:** serviços de apoio geral para auxiliar o ambiente educacional no gerenciamento de cursos e usuários, comunicação entre professores e alunos, disponibilização de conteúdos e outros materiais didáticos, adequação aos padrões educacionais dos objetos de aprendizagem; mecanismos de segurança; entre outros serviços de apoio;

O catálogo ainda possui a categoria de “Requisitos de Qualidade” que incorpora um conjunto de requisitos não-funcionais para tratar dos aspectos referentes a variados conceitos de domínios, que agregam atributos de qualidade e são importantes para qualquer tipo de sistema. Neste grupo, estão os requisitos de segurança, usabilidade, comunicação (síncrona e assíncrona), disponibilidade de acesso, multilinguagem, etc.

Em outro trabalho, Abech (2014) propõe um modelo de arquitetura *u-learning* para a adaptação e entrega de objetos de aprendizagem (OA), baseado no estilo de aprendizagem do aluno e características do dispositivo. Por exemplo, se o aluno tem o estilo de aprendizagem visual (preferência por conteúdo de vídeo), porém seu dispositivo móvel não pode executar o vídeo no formato disponível (MP4), o sistema pode adaptar esse OA convertendo formato do arquivo para outro compatível com o dispositivo (AVI) ou até mesmo extrair o áudio do vídeo se detectar que o aluno está em movimento contínuo (correndo ou andando de bicicleta).

O trabalho também apresentou a instanciação de uma aplicação, chamada EduAdapt, a partir da arquitetura proposta, que utiliza ontologias e regras para prover a melhor adaptação do OA. Além disso, o EduAdapt ainda possui um conjunto de funcionalidades e características básicas dos ambientes educacionais genéricos, referentes aos requisitos incorporados na arquitetura, tais como: *login*, disponibilização de OA, serviços gerenciais (curso, alunos, conteúdos), envio e recebimento de notificações, consulta e controle de acesso aos recursos do sistema, entre outros.

Ao final da investigação, considerando também a análise das demais fontes desse grupo (GEE) foi possível identificar as características educacionais mais comuns, abordadas na maioria dos trabalhos, e também algumas funcionalidades genéricas que parecem consistentes nos diferentes tipos de ambiente de aprendizagem investigados. Estas informações foram novamente analisadas e organizadas por categorias que representam as funcionalidades gerais e funcionalidades relacionadas às práticas educativas. A Figura 15 mostra os grupos de funcionalidades definidos em oito categorias, sendo: (1) Autoria; (2) Avaliação de Aprendizagem; (3) Comunicação; (4) Administração; (5) Adequação a padrões; (6) Documentação; (7) Disponibilização e (8) Segurança.

A categoria “Autoria” corresponde às questões relacionadas à criação de materiais didáticos em diferentes tipos de mídias, interativos ou não interativos (objeto de aprendizagem, conteúdos de texto, vídeos, provas, questionários, etc.). A categoria “Administração” trata das questões relacionadas à administração do ambiente educacional como funcionalidades para o gerenciamento de cursos, disciplinas, usuários, etc. Já as questões referentes à aplicação de provas (*online*), correção automática das avaliações dos alunos e *feedback* dos resultados, por exemplo, são questões classificadas na categoria “Avaliação de Aprendizagem”.

A Tabela 3 mostra as categorias das funcionalidades educacionais genéricas identificadas em cada fonte do grupo GEE.

Conforme a tabela (3), pode-se observar que as funcionalidades encontradas na maioria

Figura 15 – Conceitos-chave do grupo GEE

Fonte: Elaborada pelo autor.

Tabela 3 – Comparativo dos conceitos GEE

Fontes investigadas	Autoria	Avaliação	Adequação	Adm	Comunicação	Doc	Disp	Seg
Marcolino (2019)	X	X	X	X	X	X	X	X
Escudero <i>et al.</i> (2018)	X	-	-	X	-	-	X	X
Passos (2016)	-	X	X	-	-	X	X	-
Duarte Filho (2016)	X	X	X	X	X	X	X	X
Filho (2016)	X	-	-	X	-	X	X	-
Hsu <i>et al.</i> (2016)	-	X	-	X	-	-	X	X
Torres <i>et al.</i> (2016)	-	X	-	-	X	-	-	-
Fioravanti <i>et al.</i> (2010)	X	X	X	X	X	X	X	-
Moebert <i>et al.</i> (2016)	-	-	-	-	X	-	-	-
Tsiakmaki e Hartonas (2013)	-	-	-	X	-	X	-	X
Zhao e Jin (2011)	-	X	-	X	-	X	X	-
Pinheiro (2019)	-	-	-	X	-	-	X	X

Fonte: Elaborada pelo autor.

dos trabalhos pertencem às categorias de Administração e Disponibilização das informações e materiais didáticos presentes no sistema. Ambas as categorias reúnem funcionalidades importantes para garantir a eficiência de alguns serviços do ambiente educacional, tais como, disponibilizar as informações ou conteúdos de aprendizagem para os alunos com perfil apropriado.

A seguir, são apresentadas as descrições dos grupos de funcionalidades dos ambientes educacionais genéricos:

- **Autoria:** referente à construção de conteúdos digitais de aprendizagem (texto, áudio, vídeo ou imagem) e materiais de avaliação do aprendiz (prova, exercício, *Quiz*, etc.), que o ambiente deve permitir, de maneira simples e fácil para o usuário;

- **Avaliação de Aprendizagem:** trata sobre os mecanismos do sistema relacionados ao processo de avaliação do aluno, possibilitando a aplicação de provas, correções automatizadas ou feitas pelo professor, *feedback* dos resultados, entre outros;
- **Adequação a Padrões:** capacidade de se adequar a diferentes padrões educacionais, permitindo o reúso e interoperabilidade dos conteúdos e recursos didáticos;
- **Administração:** relacionado aos serviços de gerenciamento do sistema, curso, materiais didáticos e dos diferentes usuários do ambiente educacional;
- **Comunicação:** relacionado aos diferentes mecanismos de comunicação que o ambiente oferece para apoiar o processo de ensino/aprendizagem individualizado (professor e aluno) ou colaborativo entre grupos de alunos;
- **Documentação:** corresponde às informações e documentos relacionados ao curso e ao ambiente educacional que descrevam as características gerais e ajudem no entendimento desses aspectos;
- **Disponibilização:** disponibiliza as informações, documentos e conteúdos didáticos armazenados no sistema, flexibilizando o acesso e uso destes materiais;
- **Segurança:** relacionado às questões de privacidade, integridade dos dados, recuperação das informações, confiabilidade, inviolabilidade e outros aspectos de segurança que o ambiente deve oferecer.

4.2.3 Investigação do Grupo de Arquitetura Orientada a Serviços (SOA)

O último grupo de informações investigadas nesta etapa foi o grupo SOA, relacionado às informações do domínio de arquitetura orientada a serviços. Esta investigação foi importante para o desenvolvimento da proposta desta pesquisa, identificando os conceitos do domínio SOA que representam as características e funcionalidades básicas das aplicações baseadas neste estilo arquitetural.

Estes conceitos foram obtidos pela análise dos trabalhos sobre arquitetura de referência orientada a serviços para ambientes de diferentes domínios de aplicação como *m-learning*, linhas de produto de *software* (LPS), linhas de processos de negócio (LPN), teste de *software* e modelo de referência para arquiteturas SOA, e podem apoiar o desenvolvimento de arquiteturas orientadas a serviço, no tratamento de questões relacionadas ao reúso, interoperabilidade, composição de serviços, entre outros aspectos da SOA.

A OASIS (*Organization for the Advancement of Structured Information Standards*), organização para o avanço dos padrões de informação estruturada, formada por um consórcio global de colaboradores para a especificação e evolução dos padrões de interoperabilidade, definiu

um modelo de referência SOA, chamado SOA-RM (*Referência Model for Service Oriented Architecture - OASIS, 2006*), e a arquitetura de referência SOA-RAF (*Reference Architecture Foundation for Service Oriented Architecture - OASIS, 2012*).

Estes padrões buscam prover um entendimento comum sobre o estilo de arquitetura SOA e apoiar o desenvolvimento de aplicações orientadas a serviços, especificando padrões, diretrizes, relacionamentos e conceitos do domínio relacionados direta ou indiretamente aos serviços SOA, tais como:

- **Descrição de Serviço:** fornecer descrições padronizadas dos serviços para que eles possam ser descobertos e utilizados por outras aplicações e serviços;
- **Visibilidade:** são as condições necessárias para que um serviço esteja visível para os consumidores, ou seja, pré-requisitos necessários para estabelecer a interação entre provedor e consumidor (descrições sobre funcionalidades, requisitos técnicos, restrições e políticas de acesso);
- **Interação:** é a comunicação entre provedores e consumidores de serviços, por meio da troca de mensagens na rede;
- **Políticas de serviço:** representa as descrições sobre condições de uso, restrições e obrigações estabelecidas para o uso dos serviços;
- **Contrato de serviço:** é o acordo que deve ser firmado entre as partes envolvidas com o serviço (provedor e consumidor), concordando com as políticas de serviço;
- **Contexto de execução:** representa o caminho formado pelo conjunto de elementos (técnicos e de negócio) envolvidos na interação de serviços, entre provedor e consumidor;
- **Modelo de dados:** refere-se ao modelo lógico de estruturação das informações associadas ao consumo do serviço.

No trabalho de [Duarte Filho \(2016\)](#), que propõe uma arquitetura de referência orientada a serviços para o domínio dos ambientes educacionais (*m-learning*), foram definidos doze requisitos arquiteturais específicos para tratar dos aspectos relacionados ao domínio dos sistemas orientados a serviços. A partir destes requisitos, foram identificados os conceitos-chave de SOA relacionados às características e funcionalidades definidas como requisitos da arquitetura proposta.

A Tabela 4 mostra estes conceitos associados a um requisito arquitetural do grupo SOA, a título de exemplo, para demonstrar como foram identificados. Ressalta-se que os conceitos foram mapeados por mais de um requisito, conforme apresentado no trabalho de [Duarte Filho \(2016\)](#).

Tabela 4 – Relação entre conceitos e requisitos (SOA)

Conceitos-Chave (SOA)	Requisitos SOA
Descrição de serviços	Disponibilizar descrições padronizadas sobre o uso dos serviços
Publicação de serviços	Disponibilizar descrições semânticas que permitam classificar e localizar facilmente os serviços nos repositórios
Integração de serviços	Permitir a interação direta ou indireta (barramento de serviços) entre os serviços
Composição de serviços	Permitir a composição de serviços e processos de negócios
Governança e Políticas	Permitir a instanciação parcial ou total da arquitetura de referência
Qualidade de serviço	Prover mecanismos para monitorar o cumprimento de requisitos de qualidade

Fonte: Elaborada pelo autor.

Seguindo essa linha, porém aplicado no domínio da LPN, [Osshiro \(2014b\)](#) definiu a **Cambuci-LPN**, arquitetura de referência baseada em serviços para apoiar o desenvolvimento de repositórios de LPN reutilizáveis.

A arquitetura foi definida a partir da especialização da arquitetura de referência de repositórios de ativos de *software* (Cambuci), que por sua vez foi instanciada parcialmente da arquitetura *RefTEST*-SOA, arquitetura de referência orientada a serviços para ferramentas de teste de *software* ([OLIVEIRA, 2010](#))

No trabalho de [Osshiro \(2014b\)](#) foram identificadas as seguintes características da arquitetura Cambuci-LPN, relacionados especificamente ao domínio dos serviços SOA: (a) os requisitos arquiteturais e os respectivos conceitos-chave (SOA) são baseados na proposta da arquitetura de referência Cambuci e (b) as camadas de serviços relacionadas a estes conceitos são iguais as camadas definidas na arquitetura de referência *RefTEST*-SOA.

Os requisitos arquiteturais (Cambuci-LPN) foram adaptados dos requisitos da arquitetura Cambuci, enquanto os conceitos de domínio (SOA) são exatamente os mesmos conceitos abordados na referida arquitetura (Cambuci), sendo eles: (1) Descrição de serviço; (2) Publicação de serviço; (3) Interação entre serviços; (4) Composição de serviços; (5) Qualidade de serviço; e (6) Requisito geral.

As camadas de serviços da Cambuci-LPN, relacionadas a estes conceitos e instanciadas a partir da arquitetura *RefTEST*-SOA, são as seguintes:

- **Camada de Apresentação:** responsável por processar eventos e apresentar visualmente as informações. Por se tratar de uma arquitetura SOA, a camada possui três elementos: (i) Descrição do Serviço: define os tipos de dados usados nas requisições de serviços e respostas aos clientes; (ii) *Engine* de Serviços: processa as requisições cliente e converte a mensagem para o tipo de dados utilizado pelos serviços solicitados; e (iii) Controlador: trata das requisições de serviços enviadas pela *Engine* de Serviços;
- **Camada de Intermediação:** responsável pelos mecanismos que permitem a publicação,

descoberta, associação e disponibilização dos serviços do sistema, por meio dos seguintes elementos que compõem esta camada: (i) Registro de Serviços: permite a publicação e descoberta dos serviços fornecidos; (ii) Agente de Serviços: é o elemento mediador entre os serviços clientes e provedor, representando um arcabouço de transporte e roteamento das mensagens trocadas; e (iii) Escalonador de Serviço: responsável pelo processamento das requisições de serviços que tenham alguma dependência associada.

- **Camada de Orquestração:** Esta camada é exatamente igual à camada de Processo de Negócio da arquitetura Cambuci, porém, com o nome diferente (Orquestração) na Cambuci-LPN. Seu papel é permitir a construção de serviços compostos pelos já existentes; e
- **Camada de Qualidade de Serviço (QoS):** verifica o cumprimento dos requisitos de qualidade dos serviços das camadas de Orquestração, Intermediação e parte da camada de Apresentação.

O trabalho de [Marcolino \(2019\)](#) apresenta uma arquitetura de linha de produto de *software* (LPS) para o desenvolvimento de ambientes *m-learning*, baseada em serviços. A arquitetura incorpora os conceitos de serviços e microserviços para facilitar questões de reúso, integração e interoperabilidade dos serviços educacionais. Os serviços, assim como os microserviços, podem utilizar diferentes protocolos de comunicação com serviços externos e tem seus próprios repositórios de dados gerenciados separadamente. As requisições aos serviços são gerenciadas pela camada *API Gateway*, que ajuda no balanceamento de carga (*load balancer*).

Além deste, outros trabalhos também são baseados em arquiteturas de serviços REST, para tratar das questões de interoperabilidade e reúso de serviços educacionais ([ESCUADERO et al., 2018](#); [PINHEIRO, 2019](#); [BRAVO et al., 2016](#); [MONTEIRO et al., 2015](#)). A Tabela 5 apresenta o comparativo dos conceitos-chave encontrados na investigação dos trabalhos do grupo SOA. Conforme a tabela (5), foram identificados seis conceitos-chave na investigação das fontes do grupo SOA, que considerou trabalhos sobre arquitetura de referência/concreta para ambientes educacionais orientado a serviços, sendo eles: Descrição de serviço; Publicação de serviços; Interação de serviços; Composição de serviços; Qualidade de serviços; e Políticas e Contratos.

Estes conceitos foram definidos pelo consorcio OASIS como especificações da arquitetura SOA. Dentre eles, pode-se observar que a maioria dos trabalhos investigados utiliza os conceitos de **Descrição de serviço**, **Publicação de serviços**, **Interação de serviços** e **Composição de serviços**, no qual a **Descrição de serviço** foi o único conceito encontrado em todas as fontes deste grupo.

Já os conceitos **Governança** e **Qualidade de serviços** só foram encontrados em metade das fontes analisadas, mais especificamente nos trabalhos com propostas de arquitetura de referência para ambientes educacionais.

Tabela 5 – Comparativo dos conceitos SOA

Fontes investigadas	Descrição	Publicação	Governança	Composição	Integração	Qualidade
Pinheiro (2019)	X	X	-	X	X	-
Marcolino (2019)	X	X	X	-	X	X
Torres <i>et al.</i> (2016)	X	-	-	X	-	-
Duarte Filho (2016)	X	X	X	X	X	X
Osshiro (2014)	X	X	X	X	X	X
Escudero <i>et al.</i> (2018)	X	X	-	-	X	-
Monteiro (2015)	X	X	-	X	X	-
Oliveira (2011)	X	X	X	X	X	X
OASIS (2006)	X	X	X	X	X	X
OASIS (2012)	X	X	X	X	X	X

Fonte: Elaborada pelo autor.

Todos os conceitos levantados na investigação dos três grupos de informação (ULE, GEE e SOA) foram importantes para o desenvolvimento da arquitetura proposta, ajudando no processo de análise e especificação dos requisitos da arquitetura de referência (etapa RA-2), por meio de uma base de informações (conceitos-chave) relevantes para o domínio desta pesquisa e consolidadas por vários trabalhos da literatura.

Na próxima Seção (4.3), são apresentados os requisitos arquiteturais definidos na etapa RA-2, a partir das informações (características, funcionalidades e conceitos) levantadas na análise das fontes.

4.3 Estabelecimento dos Requisitos Arquiteturais (RA-2)

A segunda etapa do ProSA-RA é o passo RA-2 referente ao estabelecimento dos requisitos da arquitetura de referência, discutido nesta Seção. Os requisitos arquiteturais da MOA-RA foram levantados a partir das fontes obtidas com os mapeamentos sistemáticos, assim como, pelas características e informações identificadas nos trabalhos das pesquisas complementares (MARCOLINO, 2019; PINHEIRO, 2019; ESCUDERO *et al.*, 2018; SILVA, 2017; POPOLIN; FOFONCA, 2017; DUARTE FILHO, 2016; PASSOS, 2016; FILHO, 2016; FIORAVANTI; NAKAGAWA; BARBOSA, 2010).

Os requisitos foram definidos e classificados nos mesmos grupos de informações investigadas na etapa anterior (ULE, GEE e SOA), tendo como resultado final um total de 32 requisitos da arquitetura de referência. A seguir, são apresentados os requisitos estabelecidos em cada grupo de domínio.

4.3.1 Requisitos Arquiteturais do grupo GEE (ambientes educacionais genéricos)

Com base nas informações levantadas, foi possível identificar 32 requisitos de sistemas educacionais genéricos que foram analisados e mapeados em 8 conceitos e 11 requisitos arquiteturais do grupo GEE. A Tabela 6 mostra parte dos requisitos de sistema identificados nas fontes de investigação, para exemplificar o mapeamento de quatro conceitos-chave e respectivos requisitos arquiteturais definidos neste grupo.

Vale ressaltar que, o processo de mapeamento dos requisitos em conceitos de domínio foi realizado com base na fundamentação teórica de alguns trabalhos relacionados ao processo (PINHEIRO, 2019; DUARTE FILHO, 2016; FILHO, 2016; FIORAVANTI; NAKAGAWA; BARBOSA, 2010).

Tabela 6 – Mapeamento parcial dos requisitos de sistema, conceitos requisitos arquiteturais (GEE)

Nº	Requisitos de sistemas	Conceitos	Requisitos arquiteturais (GEE)
1	Permitir a construção de conteúdos didáticos (Texto)	Autoria	Permitir autoria de materiais didáticos e avaliação do aluno GEE-1
2	Permitir a construção de conteúdos didáticos (Vídeo)		
3	Permitir a construção de conteúdos didáticos (Áudio)		
4	Permitir a construção de conteúdos didáticos (Imagem)		
5	Permitir a construção de provas		
6	Permitir a construção de exercícios		
7	Permitir a construção de <i>quiz</i>		
8	Permitir a construção de testes de múltipla escolha		
9	Permitir a construção de questões (verdadeiro ou falso)		
10	Permitir a aplicação de provas de avaliação do aluno	Avaliação	Permitir a avaliação do aluno GEE-2
11	Permitir avaliações <i>online</i>		Fornecer <i>feedback</i> da avaliação GEE-3
12	Permitir avaliações <i>offline</i>		
13	Divulgar notas das provas		
14	Enviar resultados individuais da avaliação	Comunicação	Permitir comunicação síncrona e assíncrona (GEE-4)
15	Permitir o envio de notificações		
16	Permitir a troca de mensagens entre usuários		
17	Permitir comunicação instantânea e individual (<i>chat</i>)		
18	Permitir comunicação em grupo (fórum, sala de discussão)		

Fonte: Elaborada pelo autor.

Os requisitos de sistemas educacionais (1 a 9) deram origem ao conceito “Autoria”, por tratarem de uma funcionalidade comum relacionada à construção de materiais didáticos (conteúdos de aprendizagem (1, 2, 3 e 4) e materiais de avaliação (5, 6, 7, 8 e 9). Por esse motivo, foram mapeados em um único requisito arquitetural que incorpora os dois tipos de materiais que o ambiente permite criar, sendo o requisito arquitetural “Permitir autoria de materiais didáticos e avaliação do aluno” (GEE-1). O conceito “Comunicação” foi obtido a partir dos requisitos (15 a 18) relacionados com os mecanismos de comunicação síncrona e assíncrona utilizados no contexto educacional, que o ambiente pode oferecer com base no requisito “Permitir comunicação síncrona e assíncrona” (GEE-4).

A seguir, são apresentadas as descrições de todos os requisitos arquiteturais do grupo GEE:

- GEE-1: A arquitetura deve possibilitar o desenvolvimento de ambientes educacionais que permitam a autoria de materiais didáticos e materiais de avaliação do aluno, em diferentes tipos de mídia digital (texto, vídeo, áudio, imagem);
- GEE-2: A arquitetura deve possibilitar o desenvolvimento de ambientes educacionais que possam automatizar o processo de avaliação do aluno, fornecendo um ambiente interativo para aplicação de provas ou outras atividades avaliativas;
- GEE-3: A arquitetura deve possibilitar o desenvolvimento de ambientes educacionais que forneçam *feedback* das avaliações dos alunos;
- GEE-4: A arquitetura deve possibilitar o desenvolvimento de ambientes educacionais que permitam comunicação entre usuário/sistema e usuários entre si, de forma síncrona e/ou assíncrona;
- GEE-5: A arquitetura deve possibilitar o desenvolvimento de ambientes educacionais que possam fornecer informações/documentações do curso e do ambiente de ensino;
- GEE-6: A arquitetura deve possibilitar o desenvolvimento de ambientes educacionais que possam disponibilizar os materiais didáticos criados ou inseridos no sistema;
- GEE-7: A arquitetura deve possibilitar o desenvolvimento de ambientes educacionais com capacidade para que sejam capazes de adequar conteúdos de aprendizagem a padrões educacionais (LOM, IMS, Scorm, etc);
- GEE-8: A arquitetura deve possibilitar o desenvolvimento de ambientes educacionais que permitam o gerenciamento dos usuários, com diferentes perfis e visões do sistema;
- GEE-9: A arquitetura deve possibilitar o desenvolvimento de ambientes educacionais que permitam o gerenciamento dos cursos cadastrados;

- GEE-10: A arquitetura deve possibilitar o desenvolvimento de ambientes educacionais que permitam o gerenciamento dos conteúdos e materiais didáticos presentes no sistema;
- GEE-11: A arquitetura deve possibilitar o desenvolvimento de ambientes educacionais que possam garantir a segurança das informações e recursos didáticos presentes no ambiente, como privacidade, integridade dos dados, *backup* e mecanismos para auditoria;

4.3.2 Requisitos Arquiteturais do grupo ULE (ambientes *u-learning*)

Os requisitos arquiteturais para o domínio da aprendizagem ubíqua foram obtidos da mesma forma que os requisitos GEE, porém, analisando as informações levantadas na investigação das fontes do respectivo grupo (ULE).

A Tabela 7 apresenta os requisitos de sistemas, conceitos e os requisitos arquiteturais relacionados ao domínio da *u-learning*. Foram levantados 20 requisitos funcionais e não-funcionais dos ambientes de aprendizagem ubíqua dos quais foram mapeados em 6 conceitos de domínio e 11 requisitos arquiteturais do grupo ULE.

Tabela 7 – Mapeamento parcial dos requisitos de sistema, conceitos e requisitos arquiteturais (ULE)

Nº	Requisitos de sistemas	Conceitos	Requisitos arquiteturais (ULE)
1	Permitir o uso de dispositivos sensíveis ao contexto (GPS, sensores, RFID, etc.)	Sensibilidade ao Contexto	Obter informações do contexto (ULE-1) Gerenciar as informações do contexto (ULE-2) Descoberta de recursos (ULE-3)
2	Coletar informações do contexto relevantes para aplicação (localização, temperatura, interações do usuário, etc.)		
3	Utilizar as informações do contexto para prover os serviços certos para cada usuário		
4	Descobrir automaticamente recursos e serviços próximos (serviços, usuários, OA)		
5	Permitir a adaptação automática de serviços, conteúdos e atividades didáticas ao contexto do aluno	Adaptação ao Contexto	Adaptar ao contexto (ULE-4)
6	O sistema deve ser capaz de alterar seu comportamento para se adaptar ao contexto		
—	—	—	—
16	Permitir a construção de atividades de aprendizagem móvel sensíveis ao contexto	Autoria <i>u-learning</i>	Autoria de atividades <i>u-learning</i> (ULE-10)
17	Permitir integrar conteúdos a um contexto específico		
18	Executar alguma ação em resposta às mudanças do contexto (serviço, enviar informação, exibir conteúdo, etc.)		

Fonte: Elaborada pelo autor.

Os requisitos (1, 2, 3 e 4) relacionados com as questões de identificação, coleta, utilização das informações do contexto e descoberta de recursos e novos contextos, por meio de dispositivos integrados no ambiente, deram origem ao conceito “Sensibilidade ao Contexto” e mapearam 3 requisitos da arquitetura de referência (ULE-1, 2 e 3). O conceito “Adaptação ao Contexto” foi obtido pelos requisitos de sistema 5 e 6, que tratam respectivamente sobre a adaptação dos recursos didáticos disponíveis (serviços, conteúdos e atividades) e do próprio ambiente educacional, ao contexto da aprendizagem individual do aluno. Tais requisitos foram mapeados para um único requisito arquitetural (ULE-4).

Um dos requisitos mais importantes para a arquitetura de referência proposta neste trabalho é o requisito ULE-10 (“Autoria de atividades *u-learning*”), que permite a construção de conteúdos para atividades de aprendizagem sensíveis ao contexto. Este requisito tem uma relação de dependência direta com os requisitos GEE-1 (Autoria de materiais didáticos) e ULE-1 (Obter informações do contexto), uma vez que sua funcionalidade consiste em criar conteúdos, associá-los a um contexto (localização) e definir a ação (ou reação) que a aplicação deve executar diante deste contexto. As descrições dos requisitos ULE são apresentadas a seguir.

- ULE-1: A arquitetura deve permitir o desenvolvimento de ambientes educacionais que possam obter as informações do contexto, por meio de dispositivos integrados ao ambiente como GPS, sensor e RFID;
- ULE-2: A arquitetura deve permitir o desenvolvimento de ambientes educacionais que possam gerenciar as informações do contexto, utilizando-as para prover serviços e informações corretamente;
- ULE-3: A arquitetura deve permitir o desenvolvimento de ambientes educacionais que possam descobrir, dinamicamente, serviços e recursos próximos, importantes para o ambiente aplicar no contexto da aprendizagem do aluno (*e.g.* catálogo virtual de livros de uma biblioteca / dispositivos dos usuários próximos...);
- ULE-4: A arquitetura deve permitir o desenvolvimento de ambientes educacionais que possam adaptar conteúdos e serviços ao contexto do aluno, proativamente;
- ULE-5: A arquitetura deve permitir o desenvolvimento de ambientes educacionais que possam personalizar serviços e conteúdos ao perfil do aluno (*e.g.* multilinguagem, recomendação de conteúdos, apresentação visual, configurações da tela, volume, etc.);
- ULE-6: A arquitetura deve permitir o desenvolvimento de ambientes educacionais que possam entregar conteúdos e materiais personalizados ao contexto atual do aluno;
- ULE-7: A arquitetura deve permitir o desenvolvimento de ambientes educacionais que possam apoiar a aprendizagem *just-in-time*, permitindo o conhecimento sob medida e no momento adequado para o aluno;

- ULE-8: A arquitetura deve permitir o desenvolvimento de ambientes educacionais para apoiar a aprendizagem com mobilidade, por meio do uso de dispositivos móveis, permitindo ao aluno acessar os recursos didáticos em qualquer lugar e a qualquer hora, mesmo durante seu deslocamento;
- ULE-9: A arquitetura deve permitir o desenvolvimento de ambientes educacionais que possam ser utilizados em diferentes plataformas de *hardware* e *software*, sem necessidade de configurações adicionais, garantindo a portabilidade do ambiente de aprendizagem;
- ULE-10: A arquitetura deve permitir o desenvolvimento de ambientes educacionais que possam apoiar a autoria de atividades *u-learning*, permitindo a construção de conteúdos e atividades de aprendizagem sensíveis ao contexto para a plataforma dos dispositivos móveis;
- ULE-11: A arquitetura deve permitir o desenvolvimento de ambientes educacionais que possam notificar sobre os novos materiais didáticos disponíveis no sistema, informando automaticamente o aluno quando um novo material for construído ou descoberto pelo sistema;

4.3.3 Requisitos Arquiteturais do grupo SOA

Neste grupo, foram definidos os requisitos arquiteturais relacionados aos ambientes orientados a serviços, a partir das informações obtidas e analisadas na investigação das fontes do grupo SOA (Seção 4.2.3). As informações descrevem as características e serviços, que representam as funcionalidades do sistema, associados aos conceitos de domínio da SOA.

A Tabela 8 mostra os requisitos do domínio de serviços, conceitos da SOA e os requisitos arquiteturais de sistemas orientados a serviços (grupo SOA) definidos para a arquitetura de referência proposta.

Os requisitos de sistemas orientados a serviços (1 e 2) tratam de dois aspectos importantes da SOA, relacionados aos conceitos “Descrição” e “Publicação de Serviços”. Estes conceitos mapearam dois requisitos arquiteturais (SOA-1 e SOA-2) e são fundamentais para o contexto de serviços, pois permite que suas descrições sejam publicadas e posteriormente descobertas (serviços e desenvolvedores interessados). Além disso, ajudam em outras questões como definir e apresentar regras de contrato de serviços, fornecer as características de qualidade dos serviços, implementar os mecanismos de comunicação e facilitar o reúso, por exemplo. Conforme a Tabela 8, é possível observar que alguns requisitos arquiteturais estão relacionados a mais de um conceito como o requisito SOA-5 (“Tratamento uniforme dos serviços educacionais”). Este requisito se relaciona direta ou indiretamente com os conceitos “Descrição de Serviços”, “Publicação de Serviços” e “Integração de Serviços”. O tratamento uniforme dos serviços é um requisito importante para o conceito de integração de serviços, permitindo que os diferentes serviços possam ser descritos, publicados, descobertos e consumidos da mesma forma.

Tabela 8 – Mapeamento dos requisitos de sistema, conceitos e requisitos arquiteturais SOA

Nº	Requisitos de sistemas	Conceitos	Requisitos arquiteturais (SOA)
1	Fornecer descrições formais sobre as características técnicas dos serviços fornecidos pelo ambiente (e.g. tipo de dados e instruções)	Descrição de Serviços / Publicação de Serviços	Fornecer descrições semânticas dos serviços (SOA-1)
2	Disponibilizar informações sobre as características do ambiente, normas de uso, aspectos de qualidade, etc., utilizando descrições padronizadas		Disponibilizar informações e documentos do sistema, com descrição padronizada (SOA-2)
3	Permitir que os repositórios de serviços educacionais sejam publicados e descobertos por outras aplicações	Publicação de Serviços / Governança	Permitir a publicação e descoberta de serviços (SOA-3)
4	Permitir que novos serviços possam ser criados a partir da composição dos serviços já existentes (coreografia ou orquestração)	Composição de Serviços	Permitir a composição de serviços (SOA-4)
5	Os serviços devem ser tratados de maneira uniforme, devendo ser publicados, descobertos e consumidos da mesma forma	Descrição, Publicação e Integração de Serviços	Tratamento uniforme dos serviços educacionais (SOA-5)
6	Permitir evolução incremental do ambiente pela integração de novos módulos e serviços	Integração de Serviços	Permitir a integração de novos serviços (SOA-6)
7	Permitir que os serviços possam interagir diretamente ou via barramento, com outros serviços	Interação entre Serviços	Permitir interação direta ou via barramento de serviços (SOA-7)
8	O ambiente deve poder interagir com aplicações de plataformas heterogêneas (<i>hardware</i> e <i>software</i>)	Interação / Governança / Interoperabilidade	Interagir com aplicações heterogêneas (<i>hardware</i> e <i>software</i>) - (SOA-8)
9	Permitir o desenvolvimento de ambientes educacionais utilizando apenas parte dos módulos de serviços da arquitetura de referência	Governança	Permitir instanciação parcial (SOA-9)
10	Monitorar os serviços (cliente e servidor), em relação ao atendimento dos requisitos de qualidade definidos e disponibilizados pelo sistema (SOA-2)	Qualidade de Serviços / Governança	Permitir o monitoramento da qualidade de serviços (SOA-10)

Fonte: Elaborada pelo autor.

Os requisitos do grupo SOA são apresentados a seguir:

- SOA-1: A arquitetura deve apoiar o desenvolvimento de ambientes educacionais que forneçam descrições semânticas dos serviços, possibilitando sua classificação nos repositórios de serviços educacionais;
- SOA-2: A arquitetura deve permitir o desenvolvimento de ambientes educacionais que

possam disponibilizar informações sobre as características do ambiente, normas de uso e requisitos de qualidade, utilizando descrições padronizadas;

- SOA-3: A arquitetura deve apoiar o desenvolvimento de ambientes educacionais fornecendo mecanismos para permitir a publicação e descoberta dos serviços educacionais do ambiente, por outras aplicações (serviço cliente);
- SOA-4: A arquitetura deve prover mecanismos para apoiar a composição de serviços, a partir dos serviços já existentes no ambiente educacional;
- SOA-5: A arquitetura deve permitir o tratamento uniforme dos serviços educacionais, para que possam ser publicados, descobertos e utilizados da mesma forma;
- SOA-6: A arquitetura deve apoiar o desenvolvimento de ambientes educacionais que possam integrar novos módulos e serviços de forma gradual, permitindo a evolução incremental do ambiente;
- SOA-7: A arquitetura deve apoiar o desenvolvimento de ambientes educacionais permitindo a interação direta ou via barramento de serviços, entre os serviços (educacionais e aplicação cliente);
- SOA-8: A arquitetura deve permitir o desenvolvimento de ambientes educacionais que possam interagir com outras aplicações educacionais de plataformas heterogêneas, ou seja, desenvolvidos em diferentes linguagens de programação, sistemas operacionais e tipos de dispositivos;
- SOA-9: A arquitetura deve permitir o desenvolvimento de ambientes educacionais, a partir da sua instanciização parcial, ou seja, o ambiente pode ser construído implementando apenas parte dos módulos de serviços propostos na arquitetura de referência;
- SOA-10: A arquitetura deve permitir o desenvolvimento de ambientes educacionais que possam monitorar o atendimento aos requisitos de qualidade definidos para os serviços cliente e provedor, fornecendo mecanismos para monitorar e registrar quando algum serviço (cliente/provedor) não atender a um dos requisitos de qualidade estabelecido no ambiente educacional.

4.4 Projeto da Arquitetura de Referência (RA-3)

O projeto arquitetural corresponde à fase de especificação da arquitetura de referência, utilizando os requisitos arquiteturais estabelecidos na etapa anterior (RA-2) para construir a arquitetura e documentar o projeto arquitetural. A documentação do projeto é importante para facilitar a compreensão sobre a arquitetura de referência, pois fornece um conjunto de visões e descrições arquiteturais que representam e explicam sobre os principais aspectos da arquitetura.

Visões arquiteturais são abstrações do *design* da arquitetura que servem para representar diferentes pontos de vista do processo de construção do sistema, mostrando a organização e relacionamentos entre os componentes, elementos lógicos e de infraestrutura, entre outros. Desta forma, existe uma variedade de visões arquiteturais modeladas para tratar de diferentes questões relacionadas ao desenvolvimento da aplicação.

Baseado no modelo ProSA-RA (NAKAGAWA, 2006), a documentação da MOA-RA é formada por quatro visões arquiteturais: (1) Visão geral, (2) Visão de módulo, (3) Visão em tempo de execução e (4) Visão de implantação. Por se tratar de uma arquitetura de referência representada em alto nível de abstração, algumas visões foram desconsideradas neste trabalho, tais como a visão de implementação e visão de dados (OSSHIRO, 2014b; NAKAGAWA, 2006).

Para auxiliar a construção das visões arquiteturais, foi criado o modelo básico da estruturação e organização dos componentes da AR, desenhado em alto nível de abstração para ilustrar a visão macro da MOA-RA. O modelo representa o embrião das visões arquiteturais, modeladas com menor nível de abstração.

A Figura 16 apresenta o modelo estrutural da arquitetura proposta.

Figura 16 – Modelo estrutural da arquitetura

Fonte: Elaborada pelo autor.

Conforme a Figura 16, pode-se observar que a arquitetura é composta por três aplicações clientes e um conjunto de serviços *web* que fornecem as funcionalidades para as aplicações clientes, via conexão com a internet.

As aplicações clientes têm papéis diferentes que integram o sistema como um todo, sendo elas uma aplicação *web* (Plataforma de Gerenciamento) e duas aplicações móveis (Plataforma de Autoria e Aprendizagem).

A proposta é dividir algumas funcionalidades específicas para serem utilizadas pelos dife-

rentes tipos de usuários do ambiente educacional como, por exemplo, funcionário administrativo, professor e aluno.

As características e funcionalidades de cada aplicação são apresentadas a seguir:

- **Plataforma de Gerenciamento (PG):** Representa a principal aplicação cliente reunindo a maioria das funcionalidades do ambiente educacional oferecidas pelos serviços *web*, dando suporte para as outras aplicações, por meio de um conjunto de funcionalidades genéricas para o gerenciamento do ambiente e apoio às práticas de ensino/aprendizagem, tais como serviços de cadastro, matrícula, emissão de relatórios, comunicação, criação e disponibilização de materiais didáticos, aplicação de provas, etc. Por se tratar de uma aplicação *web* ela pode ser executada pelo navegador de qualquer computador ou dispositivo móvel. No entanto, computadores *desktop* ou notebooks são mais apropriados para utilizar a aplicação do que os dispositivos móveis que geralmente têm dimensões de tela menor (e.g. *smartphone* e *tablet*), devido à quantidade de informações e serviços gerenciados por ela. A aplicação também deve controlar o acesso ao sistema fornecendo a visão adequada para cada tipo de perfil do usuário;
- **Plataforma de Autoria:** Esta aplicação é de uso exclusivo do professor e serve para facilitar a autoria de atividades educativas sensíveis ao contexto. Trata-se de um aplicativo móvel que pode ser visto como uma plataforma genérica para a construção de conteúdos didáticos que podem reagir a um determinado contexto (localização) definido em uma atividade de aula de campo.
- **Plataforma de Aprendizagem:** É o aplicativo onde são executadas as atividades criadas pelo professor, servindo como uma plataforma de execução genérica instalada no dispositivo móvel do aluno. Além de executar as atividades, a aplicação também pode obter o contexto, criar anotações e responder exercícios propostos na atividade.

Considerando estas características e o modelo de estrutura proposto, o arcabouço foi utilizado como referência para a modelagem das visões arquiteturais da MOA-RA, assim como os requisitos definidos na etapa anterior (RA-2).

A seguir, são apresentadas as visões e descrições da arquitetura MOA-RA, na seguinte ordem: Visão geral (Seção 4.4.1); Visão de módulo (Seção 4.4.2); Visão em tempo de execução (Seção 4.4.3); e Visão de implantação (Seção 4.4.4).

4.4.1 Visão Geral

A visão geral da MOA-RA foi construída com base no modelo cliente/servidor e nos requisitos arquiteturais definidos na etapa anterior, levando em consideração o modelo estrutural do arcabouço, seguindo o estilo arquitetural SOA e o padrão de arquitetura MVC para separar a lógica de negócio da lógica de apresentação.

Esta visão é importante para compreender os aspectos gerais da arquitetura, em termos de estrutura, componentes e relacionamentos, de acordo com o modelo de estrutura discutido anteriormente. A arquitetura é apresentada na Figura 17, sendo composta por cinco camadas de serviços armazenadas no provedor *web* e três aplicações clientes.

Por se tratar de uma arquitetura baseada em serviços, que permite a comunicação entre aplicações heterogêneas, as aplicações podem ser construídas utilizando diferentes tecnologias e linguagens de programação, desde que seja compatível com a plataforma onde será executada a aplicação (*e.g.* aplicativo móvel, aplicação *web*, *desktop*).

No lado “servidor”, a arquitetura é formada por cinco camadas (Persistência (Dados), Apresentação, Aplicação, Qualidade de Serviços e Intermediação) que incorporam um conjunto de módulos de serviços para tratar de questões específicas entre si (módulo), relacionadas à sua respectiva camada. Estes módulos podem se relacionar com os módulos internos ou externos à sua camada. Vale ressaltar que as camadas de intermediação, apresentação e persistência da MOA-RA, por serem camadas independentes do domínio de aplicação da arquitetura, são semelhantes às de outras arquiteturas já validadas e consolidadas na literatura, a exemplo dos trabalhos de Duarte Filho (2016), Osshiro (2014a), Oliveira e Nakagawa (2011) e Fioravanti, Nakagawa e Barbosa (2010).

A seguir, são apresentadas as descrições sobre as camadas e módulos da arquitetura de referência MOA-RA:

Camada de Persistência (*persistence layer* ou Camada de Dados (*data layer*)): É responsável pelo armazenamento e recuperação dos dados utilizados no ambiente educacional, garantindo a persistência das informações na base de dados do sistema.

A camada fica localizada no lado servidor e possui um módulo específico para tratar do controle e integridade dos dados, chamado “módulo de Persistência”. Este módulo é responsável pelo gerenciamento do banco de dados, podendo utilizar um Sistema de Gerenciamento de Banco de Dados (SGBD) para auxiliar nas operações básicas do banco de dados como controle de acesso, armazenamento, consulta, alteração e exclusão dos dados.

Camada de Apresentação (*presentation layer*): Responsável pela apresentação visual das informações, fornecendo a interface gráfica no estilo apropriado para a aplicação que solicitou as informações. A camada é formada por três elementos para tratar dos aspectos de orquestração dos serviços solicitados pelo cliente e compatibilidade dos dados das requisições, sendo os seguintes elementos:

- **Descrição do Serviço:** define os tipos de dados usados na requisição de serviços e resultado retornado ao cliente;
- **Engine de Serviço:** processa as requisições cliente convertendo a mensagem para o tipo de dado utilizado pelos serviços da Camada de Aplicação; e

Figura 17 – Visão geral da MOA-RA

Fonte: Elaborada pelo autor.

- **Controlador:** responsável por orquestrar a execução dos serviços solicitados pela requisição

Camada de Aplicação (*application layer*): É a principal camada da MOA-RA onde estão concentrados os serviços educacionais mais importantes para a proposta da arquitetura, relacionados às principais características e funcionalidades dos ambientes de aprendizagem

sensíveis ao contexto. A camada é formada por seis módulos de serviços que tratam de questões específicas e necessárias aos ambientes do domínio educacional. Desta forma, a Figura 18 apresenta uma visão ampliada dos módulos educacionais da camada de Aplicação para facilitar a compreensão sobre os aspectos gerais desta camada, que incorpora o maior número de módulos e trata das principais questões da arquitetura.

Figura 18 – Visão ampliada dos Módulos Educacionais (MOA-RA)

Fonte: Elaborada pelo autor.

- **Módulo de Suporte:** incorpora três módulos que tratam de serviços básicos e genéricos aos ambientes educacionais, não relacionados diretamente com questões de ensino e aprendizagem, que são:
 - **Acesso:** responsável por controlar o acesso ao ambiente e aos serviços que ele oferece (e.g. *login*, recuperação de senhas, chave de acesso);
 - **Documentação:** responsável por toda a documentação disponibilizada aos usuários do sistema, fornecendo mecanismos para o armazenamento e gerenciamento da documentação técnica (manual do uso, ajuda online, informações do sistema, etc.) e educacional (plano de aula, informações do curso, registro de atividades);
 - **Comunicação:** trata dos diferentes tipos de comunicação que podem ser utilizados pelo ambiente educacional, provendo mecanismos de suporte para comunicação síncrona e assíncrona, e ferramentas colaborativas (e.g. SMS, mensagem de voz, *e-mail*, chamada de vídeo, videoconferência, fóruns, grupos, *chat* e *Wikis*).
- **Módulo de Administração:** relacionado às questões administrativas do ambiente educacional, tais como cadastro do curso/usuário, matrícula de alunos, encargos docente, relatórios, ficha do aluno, informações do curso, entre outros. Basicamente, o módulo trata das seguintes questões: (1) **Administração de curso:** responsável pelo gerenciamento das informações

- relacionadas ao curso e disciplinas; e (2) Administração de usuários: trata especificamente dos dados do usuário, como perfil, dados pessoais, permissões, disciplinas matriculadas, etc.
- Módulo de Ensino: trata das questões relacionadas à disponibilização, apresentação e entrega dos materiais didáticos (conteúdos e atividades da *u-learning*) criados no ambiente, e também é responsável pela avaliação do aluno permitindo a aplicação de provas *online*, exercícios, etc.
 - Módulo de Autoria: fornece apoio para a criação de materiais didáticos (conteúdos e avaliações) e atividades de aprendizagem. O módulo trata das seguintes questões: (1) Conteúdo: permite a criação e edição de conteúdos didáticos em diferentes tipos DE mídia (texto, imagem, áudio, vídeo) e materiais de avaliação *online* (prova, exercício interativo, questionário, teste de múltipla escolha, etc); (2) Atividade *context-aware*: permite a criação de conteúdos para atividades sensíveis ao contexto da localização (aplicações *u-learning*), ou seja, as atividades utilizam o conteúdo criado anteriormente (*e.g.* texto, questionário, imagem), integrado de alguma forma com os dados de localização (coordenadas de latitude e longitude) obtidos no Módulo de Gerência do Contexto para criar o conteúdo sensível ao contexto (arquivo) que será disponibilizado e executado no aplicativo da atividade, que corresponde à aplicação cliente do aluno (plataforma de aprendizagem); e (3) Captura de conteúdo: responsável pela obtenção e armazenamento das interações do aluno durante a execução da atividade. Serve como um mecanismo que pode ser inserido na construção do conteúdo, avaliação ou atividade para registrar as interações e respostas fornecidas pelo aluno.
 - Módulo de Gerência do Contexto: trata das questões relacionadas com aquisição de informações do contexto, mais especificamente ao contexto da localização geográfica (coordenadas de latitude e longitude – GPS), descoberta automática do contexto e serviços baseados na localização ou LBS (*location based services*). O módulo é importante para a proposta da arquitetura, pois representa um elemento fundamental para a composição e definição do comportamento das atividades ubíquas, disponibilizando um conjunto de LBS que podem ser integrados nas funcionalidades destas atividades, como cálculo de proximidade, distância, serviço de mapa, trajeto do usuário, etc.
 - Módulo de Adaptação: responsável por mecanismos que permitem a adaptação do ambiente às características individuais do usuário ou do contexto onde está ocorrendo a aprendizagem. Baseado no perfil do aluno ou preferências de estudo, o ambiente pode passar a recomendar materiais mais adequados às suas características, ou ainda, ajustar as configurações do dispositivo para adaptar-se ao contexto do ambiente externo (*e.g.* exibir somente conteúdos de áudio ou vídeo se o usuário estiver em movimento, ajustar o brilho de acordo com a luminosidade do local, entre outros).

Vale destacar os relacionamentos entre os principais módulos da camada de aplicação (autoria, ensino e contexto), que ocorrem da seguinte forma: o módulo de autoria usa serviços do módulo de contexto para criar atividades *context-aware*, e com o módulo de ensino que disponibiliza as atividades aos alunos.

Camada de Qualidade de Serviço (*Quality of Service Layer*): responsável por verificar se as demais camadas provedoras de serviços estão cumprindo os requisitos de qualidade estabelecidos, atuando diretamente sob a camada de intermediação e parte da camada de apresentação (descrição dos serviços).

Camada de Intermediação (*intermediation layer*): Responsável por controlar as requisições enviadas entre serviços clientes e provedor, e fornecer mecanismos para a descoberta, publicação e disponibilização dos serviços educacionais do ambiente, entre outros. A camada é composta pelos seguintes elementos:

- **Registro de Serviços:** responsável pela publicação das informações dos serviços do ambiente, permitindo que possam ser descobertos e utilizados por outros serviços ou aplicações. Pode ser concretizado por meio de *brokers* ou *matchmakers*;
- **Agente de Serviço:** é o elemento mediador da comunicação entre cliente e servidor, atuando como arcabouço de roteamento e transporte das requisições aos serviços disponibilizados no ambiente educacional. O Agente de Serviço pode ser efetivado por meio de um ESB;
- **Escalonador de Serviço:** verifica e processa as pendências das requisições aos serviços feitas pela aplicação cliente, descritas por uma linguagem padrão (*e.g.* XML). No caso das requisições a serviços compostos, o Escalonador de Serviços analisa também suas políticas de uso e requisitos de interoperabilidade necessários para a composição dos serviços solicitados.

Por fim, o conjunto de serviços ortogonais são serviços transversais que podem se comunicar com todos os serviços dos demais módulos. Tais serviços estão relacionados à persistência, segurança, configuração, responsividade, entre outros. O serviço de Segurança tem um papel importante para qualquer tipo de ambiente, fornecendo mecanismos que ajudam a controlar o acesso aos serviços e registrar as atividades de cada seção do usuário para, caso necessário, possibilitar auditoria no sistema (autenticação, *logs* e registro de atividades).

4.4.2 Visão de Módulos

A visão de módulo é utilizada para representar a arquitetura de referência em relação à forma como os módulos estão organizados e relacionados entre si, por meio de pacotes, subpacotes, classes e relacionamentos, fornecendo uma visão ampla dos elementos que implementam as funcionalidades dentro de cada módulo que compõe a arquitetura.

A Figura 19 mostra a visão de módulo da MOA-RA, descrita em UML e construída a partir dos requisitos arquiteturais (Seção 4.3), visão geral e foi parcialmente baseada nas visões de módulo de outras arquiteturas de referência como *Ref-mLearning* (DUARTE FILHO, 2016), Cambuci (OSSHIRO, 2014a) e *RefTEST-SOA* (OLIVEIRA; NAKAGAWA, 2011), devido à abordagem orientada a serviços, comum a todas.

É importante lembrar que os elementos da visão de módulos são os mesmos elementos da visão geral da MOA-RA. Por esse motivo, as descrições da visão de módulos foram mais direcionadas nos grupos de serviços de cada pacote representado nesta visão, mesmo assim ainda pode haver certa redundância com as descrições da visão apresentada anteriormente (Seção 4.4.1).

Os módulos e sub-módulos das camadas são representados por pacotes e sub-pacotes, que podem ser refinados em classes de objetos e interfaces das funcionalidades contidas em cada sub-módulo das camadas. Os pacotes e sub-pacotes identificados em cada camada da MOA-RA e representados na visão de módulos, são apresentados na Tabela 9.

Os serviços relacionados à camada de Intermediação são representados pelos pacotes *RegistroServiço* e *MediadorServiço*. O pacote *RegistroServiço* é responsável pela publicação e localização dos serviços utilizados no ambiente educacional. Já o pacote *MediadorServiço* é composto por dois grupos de serviços, cujo objetivo é tornar possível as interações entre os serviços clientes e os serviços educacionais da camada de aplicação da MOA-RA, atuando como arcabouço de roteamento e transporte (pacote *AgenteServiço*), e também verificando e processando as pendências das requisições dos serviços (pacote *EscalonadorServiço*).

Os serviços do pacote de Apresentação são responsáveis pelo processamento das requisições de serviços, convertendo seus dados para o tipo de dados utilizado pelos serviços, e pela gerência dessas requisições de serviços, por meio dos seus respectivos sub-pacotes: *EngineServiço* e Controlador.

Os serviços relacionados ao domínio dos ambientes de aprendizagem ubíqua, tratados pelos módulos educacionais da camada de Aplicação (Visão Geral), estão presentes em seis grupos de pacotes e seus respectivos sub-pacotes de serviços educacionais representados na visão de módulo da MOA-RA. Estes serviços estão agrupados nos seguintes pacotes:

- Pacote de Suporte: composto por três grupos de serviços de apoio ao ambiente educacional, classificados nos seguintes sub-pacotes:
 - Acesso: serviços responsáveis por fornecer e controlar o acesso dos usuários no ambiente educacional (*e.g.* gerador de chave de acesso, senhas, *login*, etc.);
 - Comunicação: serviços de comunicação utilizados pelo sistema (*e.g.* *e-mail*, mensagens, *chats*, notificações, etc.);

Tabela 9 – Pacotes e sub-pacotes da visão de Módulos (MOA-RA)

Camada	Pacote	Sub-pacote
Intermediação	<i>RegistroServiço</i> <i>MediadorServiço</i>	<i>AgenteServiço</i> <i>EscalonadorServiço</i>
Apresentação	<i>EngineServiço</i> Controlador	
Aplicação	Suporte	Acesso Comunicação Documentação
	Administração	<i>AdministraçãoUsuário</i> <i>AdministraçãoCurso</i>
	Ensino	Avaliação Entrega Apresentação
	Autoria	Conteúdo Atividade <i>CapturaContexto</i>
	<i>GerênciaContexto</i>	Aquisição Descoberta <i>ServiçoLocalização</i>
	Adaptação	<i>AdaptaçãoUsuário</i> <i>AdaptaçãoContexto</i>
Persistência	<i>ServiçosOrtogonais</i>	Persistência
Qualidade de Serviço	Qualidade	

Fonte: Elaborada pelo autor.

- Documentação: serviços responsáveis por gerenciar e disponibilizar toda a documentação do ambiente (*e.g. help*, manual do usuário, documentos acadêmicos, entre outros);
- Pacote de Administração: formado por dois grupos de serviços responsáveis por gerenciar as informações de apoio para:
 - Avaliação: serviços de avaliação dos alunos, permitindo a aplicação de provas interativas, de forma automatizada ou semi-automatizada (*e.g. provas online*, exercícios interativos, *Quiz*, etc.);
 - Entrega: serviços de entrega automática dos novos materiais didáticos, criados ou inseridos no ambiente educacional, aos usuários endereçados (*e.g. link* do material, *e-mail* anexado);
 - Apresentação: serviços associados à apresentação e disponibilização dos materiais didáticos, no ambiente educacional (*e.g. quadro de notas*, página de apresentação, *microblogs*, exibição e acesso aos materiais);

- Pacote de Ensino: dividido em três sub-pacotes de serviços relacionados às questões de apoio ao ensino:
 - Aquisição: serviço responsável por coletar as informações do contexto, enviadas por diferentes dispositivos e tecnologias de captura, integrados no ambiente educacional (*e.g.* GPS, leitor *QR-Code*, sensor);
 - Descoberta: serviço que monitora as mudanças do contexto e identifica um novo contexto no raio de cobertura do dispositivo utilizado, ajudando a prover as adaptações necessárias;
 - *ServiçosLocalização*: relacionado ao conjunto de serviços baseados na localização (LBS), que utiliza os dados de localização do usuário para fornecer diferentes funcionalidades como cálculo de distância, proximidade, rota percorrida, mapa, etc. Estes serviços podem ser implementados ou reutilizados (*e.g.* *Google maps*);
- Pacote de *GerênciaContexto*: composto por três sub-pacotes de serviços responsáveis pelo tratamento e gerenciamento das informações do contexto, que podem ser utilizadas pelo ambiente educacional para apoiar a aprendizagem:
 - *AdaptaçãoUsuário*: adaptação baseada nas informações do contexto do usuário (*e.g.* perfil, preferências, *log* de acesso, interações). O serviço pode utilizar ontologias disponíveis e adequadas ao contexto em questão, para auxiliar na implementação desse serviço;
 - *AdaptaçãoContexto*: esse outro grupo de serviços trata da adaptação ao contexto físico no qual ocorre a aprendizagem, considerando a localização do usuário, dia e horário, luminosidade, entre outras informações do ambiente físico que possam ser utilizadas para prover as adaptações necessárias ao sistema.

Adicionalmente, outros pacotes de serviços também estão representados na visão de módulo da MOA-RA, sendo: (1) Pacote Cliente, incorpora os serviços das aplicações clientes para representar como os usuários podem consumir os serviços disponíveis no ambiente educacional; (2) Pacote *QualidadeDeServiço*, relacionado aos serviços de monitoria e sinalização do não atendimento dos requisitos de qualidade, definidos e disponibilizados no ambiente educacional (descrição dos serviços); e (3) Pacote de *ServiçosOrtogonais*, relacionado ao conjunto de serviços de apoio geral ao ambiente educacional, fornecendo uma variedade de funcionalidades para tratar de questões de segurança, persistência, configurações do sistema e dispositivos, entre outros serviços.

Por fim, vale lembrar que por se tratar de uma arquitetura de referência baseada no estilo SOA, os serviços de cada pacote que compõe a visão de módulo da MOA-RA devem ser construídos, levando em consideração os princípios da SOA como baixo acoplamento,

4.4.3 Visão em Tempo de Execução

A visão em tempo de execução da MOA-RA mostra a estrutura do ambiente educacional em tempo de execução, ou seja, durante o funcionamento do sistema, representando graficamente seus principais componentes e os relacionamentos entre eles, as interfaces fornecidas e consumidas, protocolos de comunicação, entre outros aspectos que se manifestam durante a execução do sistema. Os componentes incorporados na visão em tempo de execução da MOA-RA representam os pacotes de serviços da visão de módulos.

Os protocolos e linguagens utilizadas por serviços *web* e representadas na visão em tempo de execução da MOA-RA foram inseridos para ilustrar essas questões, buscando facilitar a compreensão sobre a arquitetura. No entanto, é importante esclarecer que a arquitetura proposta é independente de tecnologias de implementação, permitindo que seus componentes possam interagir utilizando outros protocolos e tecnologias equivalentes.

A visão em tempo de execução da MOA-RA foi construída com base na visão em tempo de execução da arquitetura de referência *Ref-mLearning* (DUARTE FILHO, 2016) que tem proposta parecida com a MOA-RA, porém, aplicada apenas no domínio da aprendizagem móvel. A Figura 20 apresenta a visão em tempo de execução da MOA-RA na qual é possível observar duas formas em que o serviço cliente (componente Cliente) pode interagir com o ambiente para consumir os serviços educacionais, representados pelos componentes: Acesso, Comunicação, Documentação, Administração, Ensino, Autoria, *GerênciaContexto* e Adaptação. A primeira, mostra as interações e componentes que o serviço cliente deve fazer quando não souber o endereço do serviço provedor. No segundo caso, o consumidor já sabe o endereço e as informações do serviço necessárias para se comunicarem.

Quando o serviço cliente desconhece o endereço do serviço que vai utilizar, o primeiro passo é procurar pelo serviço em um registro de serviços, enviando uma requisição para o serviço responsável (componente *RegistroServiço*), que pesquisa na base de dados (componente *RepositórioServiço*) e, se localizar, o serviço *RegistroServiço* envia o endereço como resultado da pesquisa. A comunicação entre estes serviços deve ser feita utilizando um protocolo padrão, ilustrado na Figura 20 com o protocolo UDDI. Em seguida, o serviço cliente estabelece conexão com o serviço desejado e solicita a descrição do respectivo serviço para que possam se comunicar corretamente. É importante que as descrições dos serviços sejam uniformes, seguindo um padrão de escrita e estrutura que possa descrever a semântica de cada serviço, para facilitar no seu entendimento, utilização, classificação e localização no repositório de serviços.

Desta forma, deve-se utilizar uma linguagem padrão para descrever os serviços como, por exemplo, WSDL. Ao final, os serviços (cliente e provedor) podem se comunicar utilizando um protocolo padrão como o SOAP, por exemplo.

Caso o endereço e a descrição do serviço provedor já sejam conhecidos pelo serviço consumidor, eles podem estabelecer comunicação direta ou indireta, por intermédio dos servi-

Figura 20 – Visão em Tempo de Execução (MOA-RA)

Fonte: Elaborada pelo autor.

ços do componente *AgenteServiço*, utilizando os mesmos padrões de linguagem e protocolo exemplificados no primeiro caso (WSDL e SOAP).

O componente *QualidadeDeServiço* é responsável por monitorar o cumprimento dos requisitos de qualidade do ambiente educacional (requisitos não-funcionais), durante a comunicação entre os serviços cliente e provedor, fornecendo interface para todos os componentes de

serviços de intermediação e parte dos serviços de apresentação.

Por fim, pode-se observar que todos os serviços educacionais podem consumir as funcionalidades dos serviços ortogonais, uma vez que o componente Ortogonais, que representa os respectivos serviços, fornece a interface para todos os componentes dos serviços educacionais. Os serviços ortogonais são relacionados aos serviços de persistência, segurança e outros serviços transversais, com funcionalidades de apoio para o ambiente educacional.

4.4.4 Visão de Implantação

A visão de implantação da MOA-RA (Figura 21) representa a estrutura física da arquitetura, em relação à distribuição, organização e conexões entre os equipamentos de *hardware* que compõem toda a infraestrutura do ambiente educacional. Esta visão ajuda a compreender a distribuição física dos serviços, ilustrando as relações entre a parte lógica (elementos de *software*) e a parte física (elementos de *hardware*) do ambiente educacional. Isso permite entender quais módulos de serviços são executados em quais equipamentos de *hardware*, e como esses equipamentos estão conectados, uma vez que a MOA-RA é uma arquitetura orientada a serviços dos quais podem estar alocados em máquinas diferentes, interagindo de forma distribuída.

Vale ressaltar que a visão de implantação da MOA-RA tem aspectos semelhantes às visões de implantação da *Ref-mLearning* (DUARTE FILHO, 2016) e Cambuci (OSSHIRO, 2014b), em relação à infraestrutura física e distribuição dos componentes de *hardware* e *software*, por se tratar de arquiteturas de referência também orientadas a serviços. Os elementos que compõem a visão de implantação da MOA-RA são apresentados a seguir:

- **Cliente Web:** representa o dispositivo do usuário utilizado para acessar o ambiente educacional, por meio do navegador web instalado na máquina cliente;
- **Cliente Móvel:** representa exclusivamente o dispositivo móvel do usuário utilizado para acessar o ambiente educacional (*tablet*, *smartphone*, etc.);
- **Servidor Web:** responsável por responder às requisições dos serviços clientes (navegador e app) e fornecer acesso aos serviços do ambiente educacional (e.g. *Tomcat*);
- **Servidor de Aplicação:** contém todos os serviços educacionais do ambiente (autoria, ensino, gerência do contexto, adaptação, entre outros) e também pode conter uma *Engine* de serviços (e.g. *AXIS2*). O Servidor de Aplicação (e.g. *JBoss*) é responsável por processar as requisições enviadas pela aplicação cliente, utilizando protocolos SOAP e WSDL, por exemplo;
- **Servidor de Agente de Serviço:** armazena os serviços intermediários, representados pelos componentes *AgenteServiço* e *EscalonadorServiço*, podendo ser implementado por um ESB;

serviços, que podem ser gerenciados utilizando um SGBD (Sistema de Gerenciamento de Banco de Dados), por exemplo, *MySQL*.

4.5 Avaliação da Arquitetura de Referência (RA-4)

A última etapa do ProSA-RA corresponde ao processo de avaliação da arquitetura de referência (passo RA-4) para verificar os aspectos de qualidade da arquitetura proposta. Neste trabalho, foram utilizados dois métodos para avaliar a MOA-RA: (1) Avaliação por inspeção, utilizando um *checklist* comparativo com o modelo de referência RAModel (NAKAGAWA; FLÁVIO; BECKER, 2012); e (2) Avaliação com especialistas, aplicando questionários a dois grupos de colaboradores com experiência nas áreas da educação e arquitetura de *software*.

Para complementar o processo de avaliação da MOA-RA também foram realizados dois estudos de caso para verificar a viabilidade da aplicação prática da arquitetura de referência, por meio de sua instanciação em cenários distintos.

As avaliações são apresentadas, na íntegra, com as discussões dos resultados obtidos em cada uma, no Capítulo 5. Já as discussões sobre os estudos de caso realizados com a MOA-RA são apresentados no Capítulo 6.

4.6 Considerações Finais

Este capítulo apresentou a MOA-RA, que consiste em uma arquitetura de referência para ambientes de aprendizagem ubíqua, baseada no estilo SOA. A arquitetura foi proposta para auxiliar o desenvolvimento de ambientes educacionais que possam utilizar os recursos e benefícios provenientes das tecnologias da computação móvel e ubíqua, em especial a computação sensível ao contexto, a favor do processo de ensino-aprendizagem.

O desenvolvimento da MOA-RA foi baseado nas etapas do processo ProSA-RA, modelo de referência para o estabelecimento de arquiteturas de referência. Na primeira etapa (passo RA-1) foi feita a investigação das fontes de informação relacionadas ao domínio da MOA-RA, investigando três grupos de informações do escopo da MOA-RA, referentes aos ambientes educacionais genéricos (GEE), ambientes *u-learning* (ULE) e arquitetura orientada a serviços (SOA).

Na etapa seguinte (RA-2), foram definidos os requisitos da MOA-RA com base nas informações obtidas na investigação das fontes. As funcionalidades e requisitos de sistema levantados nas fontes foram classificados em conceitos-chave e mapeados para os requisitos arquiteturais da MOA-RA, referente a cada grupo de informação.

O projeto arquitetural foi realizado na etapa RA-3, na qual a MOA-RA foi estabelecida e devidamente documentada por meio das suas visões arquiteturais e respectivas descrições.

A MOA-RA foi representada por meio de quatro tipos de visões arquiteturas para ilustrar a arquitetura sob diferentes pontos de vista, sendo: (1) Visão Geral; (2) Visão de Módulo; (3) Visão em Tempo de Execução; e (4) Visão de Implantação.

Por fim, na etapa RA-4, que corresponde à avaliação da arquitetura de referência, foram feitas as seguintes avaliações da MOA-RA: (1) Avaliação comparativa com o modelo de referência ProSA-RA, por meio de inspeção por *checklist*; (2) Avaliação com especialistas da área; e (3) Estudo de caso, como forma de avaliação complementar, de natureza prática e aplicada. As avaliações da MOA-RA são apresentadas nos Capítulos 5 e 6.

AVALIAÇÃO DA ARQUITETURA DE REFERÊNCIA MOA-RA

5.1 Considerações iniciais

Este capítulo descreve sobre os processos utilizados na avaliação da arquitetura de referência MOA-RA, referente à etapa (RA-4) final do processo ProSA-RA.

Avaliação arquitetural é um processo baseado em métodos e instrumentos utilizados para averiguar diferentes questões relacionadas à arquitetura, tais como qualidade, incorporação de elementos importantes, viabilidade prática, consistência da documentação com a proposta da arquitetura, entre outros.

No entanto, avaliar uma arquitetura de software é uma tarefa difícil e demorada, pois necessita de planejamento, disponibilidade dos avaliadores e tempo para aplicar, coletar e analisar os resultados (NAKAGAWA, 2006). Além do mais, é importante escolher o método de avaliação mais adequado para os objetivos da avaliação desejada.

Apesar de existirem vários métodos distintos para avaliar arquiteturas de *software* aplicadas em diferentes domínios, a literatura é carente de métodos formais de avaliação para arquiteturas de referência (AR) por completo. Existem apenas métodos para avaliar partes específicas da arquitetura de referência, tais como documentação, modelagem, prototipação, etc. (DUARTE FILHO, 2016).

Considerando a falta de instrumentos para realizar uma avaliação completa dos aspectos gerais das arquiteturas de referência, foram utilizados alguns métodos específicos para avaliar a MOA-RA com objetivo de tornar sua avaliação mais completa e abrangente. Desta forma, a arquitetura foi avaliada com base nos seguintes métodos, conhecidos e consolidados na literatura da área: (1) Avaliação por inspeção, utilizada para checar um conjunto de itens importantes que a arquitetura deve incorporar; (2) Avaliação com especialistas, por meio de questionário aplicado

a grupo de especialistas da área; e (3) Estudo de caso, para verificar a viabilidade prática da arquitetura em um cenário real de aplicação, em termos de instanciação de arquiteturas concretas e, se for o caso, implementação de sistemas ou protótipos com base na AR.

Neste capítulo, são apresentadas apenas as duas primeiras avaliações realizadas com a MOA-RA (avaliação por inspeção e com especialistas). O estudo de caso é apresentado no capítulo seguinte (Capítulo 6).

O restante do capítulo está organizado de seguinte forma: A Seção 5.2 trata dos aspectos da avaliação por inspeção da MOA-RA. A Seção 5.3 refere-se à avaliação realizada com ajuda dos especialistas do domínio. Por fim, a Seção 5.4 apresenta as considerações finais do processo de avaliação da arquitetura MOA-RA.

5.2 Avaliação por Inspeção

Avaliação por inspeção é uma técnica bastante utilizada para verificar um conjunto de itens de um determinado grupo, utilizando uma lista dos itens que serão checados (*checklist*). A inspeção por *checklist* ajuda a controlar atividades, identificar falhas e detectar problemas relacionados a um determinado item, permitindo a tomada de decisões para tratar tais problemas.

Checklist é uma lista de verificação contendo um conjunto de itens previamente estabelecidos, servindo como um instrumento de controle sistemático de produtos, serviços, componentes, entre outros tipos de itens. No campo da arquitetura de *software*, mais especificamente da arquitetura de referência, este instrumento pode ser utilizado para verificar aspectos importantes relacionados a diferentes questões da arquitetura como requisitos, atendimento a padrões, incorporação de atributos de qualidade, modelagem, etc.

Desta forma, para verificar a presença do conjunto de elementos considerados importantes e necessários a qualquer arquitetura de referência, proposto no modelo RAModel (NAKAGAWA; FLÁVIO; BECKER, 2012), foi elaborado um *checklist*, baseado nos elementos do RAModel, para conduzir a avaliação por inspeção da MOA-RA.

O RAModel foi adotado nesta avaliação por se tratar de um modelo de referência completo, acessível, consolidado na literatura e também por ser independente do domínio de aplicação da arquitetura, incorporando todos os elementos necessários para uma arquitetura de referência, classificados nos seguintes grupos:

1. **Grupo de Domínio (*Domain*):** elementos relacionados às questões do domínio de aplicação da arquitetura, sendo eles: Legislação; Atributos de qualidade; e *Compliance* do sistema;
2. **Grupo de Aplicação (*Application*):** elementos que ajudam a entender diferentes aspectos da arquitetura de referência, relacionados às questões de: Restrições; Domínio dos dados;

Requisitos; Objetivos; Limitações; Riscos e Escopo da AR;

3. **Grupo de Infraestrutura:** elementos que representam a infraestrutura física e lógica utilizados na construção do sistema como: Diretrizes e boas práticas; Estrutura geral; e os Elementos físicos e lógicos do sistema;
4. **Grupo dos Elementos Transversais (*Crosscutting Elements*):** elementos que podem se relacionar com os elementos dos demais grupos do modelo, tratando de questões como: Decisões; Terminologias de domínio; Comunicação externa e interna do sistema.

Nesta avaliação, com o auxílio do *checklist*, foram inspecionados 18 itens da MOA-RA, referente aos elementos do RAModel.

A Tabela 10, mostra o resultado da MOA-RA, na avaliação comparativa com o modelo RAModel.

Tabela 10 – Resultado da avaliação por inspeção (*Checklist*)

Grupos	Elementos RAModel	Elementos MOA-RA
Domínio	Legislação, Padrões e Regulamentos	Ok
	Atributos de qualidade	Ok
	<i>Compliance</i> do sistema	—
Aplicação	Restrições	Ok
	Domínio de dados	Ok
	Requisitos funcionais	Ok
	Objetivos e necessidades	Ok
	Limitações	Ok
	Riscos	—
	Escopo	Ok
Infraestrutura	Boas práticas e Diretrizes	Ok
	Estrutura Geral	Ok
	Elementos Físicos	Ok
	Elementos Lógicos	Ok
Elementos transversais	Decisões	—
	Terminologia do domínio	Ok
	Comunicação externa	Ok
	Comunicação interna	Ok

Fonte: Elaborada pelo autor.

De acordo com a Tabela 10, é possível observar que a MOA-RA incorpora a maioria absoluta dos elementos propostos no RAModel, contendo 15 dos 18 itens presentes no *checklist*, ou seja, a MOA-RA possui 83% dos elementos necessários para arquiteturas de referência.

A seguir, são apresentadas as discussões e análise dos resultados obtidos na avaliação de cada grupo de elementos.

5.2.1 Análise dos resultados da avaliação (inspeção)

- **Grupo de Domínio:** O elemento arquitetural relacionado às questões de legislação, padrões e regulamentos do domínio de aplicação está incorporado na MOA-RA, uma vez que a arquitetura foi construída levando em consideração os padrões, normas e recomendações existentes para o domínio das arquiteturas de referência e SOA (ex. ProSA-RA, OASIS e RAModel), assim como as diretrizes do MEC (2016) para ambientes virtuais de aprendizagem e da UNESCO (2013) para a modalidade da aprendizagem móvel. No entanto, não foram encontradas normas e diretrizes específicas para o domínio da aprendizagem ubíqua no momento em que as pesquisas foram realizadas.

Alguns atributos de qualidade também estão incorporados na MOA-RA, identificados pelos requisitos não-funcionais (e.g. GEE-7, GEE-11, ULE-2, ULE-4, ULE-5, ULE-7, ULE-8, SOA-2, SOA-6, SOA-8, SOA-10) que foram definidos de acordo com normas, modelos e diretrizes específicas para o domínio de aplicação, ajudando a garantir a presença de atributos de qualidade na MOA-RA e, conseqüentemente, nas arquiteturas concretas construídas a partir da sua instanciação.

A única deficiência da MOA-RA, em relação ao grupo de Domínio, é a ausência de elementos para tratar dos aspectos de *Compliance* do sistema devido à carência de normas e diretrizes específicas para o domínio da aprendizagem ubíqua, o que impossibilita verificar se a arquitetura de referência permite a instanciação de ambiente *u-learning* em total conformidade com a legislação e regulamentos exigidos para o domínio em questão.

- **Grupo de Aplicação:** Neste grupo, o único elemento não incorporado na MOA-RA, no momento em que esta avaliação foi realizada, refere-se ao tratamento dos riscos associados à arquitetura de referência. No entanto, após detectar esta deficiência foi feito um levantamento dos riscos mais comuns para o contexto da arquitetura de referência, SOA e ambientes educacionais genéricos, uma vez que, a aprendizagem ubíqua ainda é uma área emergente e novos riscos podem aparecer e desaparecer com o advento de novas tecnologias.

Os demais elementos deste grupo estão presentes na arquitetura avaliada, identificados pelo conjunto de requisitos funcionais do domínio da aplicação, definidos com base em investigações da literatura e incorporados em suas visões arquiteturais (geral e de módulos) e seus respectivos relacionamentos. O objetivo e escopo da arquitetura foram bem definidos e são demonstrados de forma clara no corpo da documentação arquitetural, por meio das descrições e modelagens da MOA-RA. Além disso, o documento também descreve sobre as restrições da arquitetura de referência, em termos de padrão, camadas, módulos, serviços e comunicação.

Em relação ao domínio de dados, a MOA-RA especifica diferentes domínios para o conjunto de dados utilizados pelo sistema. Por exemplo, dados do usuário, administrativo,

educacional, comunicação e contexto.

Por fim, a arquitetura MOA-RA foi projetada para apoiar o desenvolvimento de ambientes sensíveis ao contexto da localização, baseado em LBS. Tal característica representa sua maior limitação no momento. No entanto, essa limitação pode ser eliminada com base nas características do estilo SOA que permite à arquitetura evoluir integrando novos módulos e serviços.

- **Grupo de Infraestrutura:**

Todos os elementos do grupo de infraestrutura propostos pelo modelo RAModel foram detectados na arquitetura MOA-RA, por meio de um conjunto de boas práticas utilizadas no desenvolvimento, descrições e modelagens de diferentes visões arquiteturais.

Além dos padrões e diretrizes adotadas no projeto, também foram realizados mapeamentos sistemáticos da literatura, avaliações formais, informais e estudos de caso práticos, baseado na instanciação arquitetural e implementação de protótipos de ambientes *u-learning*, a partir da MOA-RA, demonstrando o uso de boas práticas e diretrizes.

As visões arquiteturais foram modeladas com ferramentas UML, para representar graficamente a estrutura geral da MOA-RA e os aspectos de *hardware* e *software* da arquitetura. Esses elementos são tratados pelo conjunto de visões e suas respectivas descrições que ajudam a entender diferentes aspectos da MOA-RA relacionados à sua estrutura geral, comportamento, aspectos lógicos e tecnológicos, entre outros. Neste sentido, a arquitetura apresenta as seguintes visões: (i) Visão geral; (ii) Visão de módulos; (iii) Visão em tempo de execução; e (iv) Visão de implantação.

- **Grupo de Elementos Transversais:**

Os elementos de comunicação interna e externa estão incorporados na MOA-RA, representados nas modelagens da visão de tempo de execução e visão de implantação. Tais visões mostram os relacionamentos internos da arquitetura, referente à comunicação entre serviços do mesmo módulo, entre serviços de módulos e/ou camadas diferentes da MOA-RA, assim como os mecanismos de comunicação com serviços externos a arquitetura (de outras aplicações). Estes elementos são descritos em termos de padrões, regras e protocolos de comunicação que os serviços devem estabelecer como acordo (*e.g.* SOAP, WSDL, UDDI, EBS).

A MOA-RA foi construída utilizando um conjunto de termos de domínio comuns às áreas incorporadas no seu escopo, ou seja, terminologias da arquitetura de referência, SOA e aprendizagem ubíqua, facilitando assim a compreensão e entendimento dos aspectos abordados nas suas descrições arquiteturais.

O ponto negativo da MOA-RA em relação ao grupo de elementos transversais é a falta de descrições formais sobre as decisões de projeto, apresentando alternativas que ajudem

a evitar os possíveis conflitos de interesses que possam ocorrer na instanciação das arquiteturas concretas e auxiliar nas tomadas de decisões no nível tecnológico.

5.3 Avaliação com Especialistas

Embora a técnica de inspeção por *checklist* possa revelar, implicitamente, alguns indícios de qualidade da arquitetura, demonstrando que ela incorpora todos os elementos necessários para arquiteturas de referência, independentemente do domínio de aplicação, essa técnica consiste em uma avaliação quantitativa. Porém, os resultados podem demonstrar indícios de qualidade, uma vez que, a quantidade de elementos importantes para as arquiteturas de referência identificado nos resultados pode sinalizar uma boa ou má qualidade da arquitetura.

Para verificar os aspectos de qualidade presentes na MOA-RA foi feita uma avaliação de natureza qualitativa (com aspectos também quantitativos), por meio de questionários aplicados a dois grupos de colaboradores experientes nas seguintes áreas do conhecimento: (1) Especialista em arquitetura de *software* e (2) Especialista do domínio (educacional).

5.3.1 Avaliadores (grupos de especialistas)

Os especialistas que participaram da avaliação da MOA-RA são profissionais que atuam na área da educação ou na indústria, com boa experiência nas atividades de ensino, pesquisa e desenvolvimento de arquiteturas e ambientes educacionais (*u-learning*, *m-learning* ou *e-learning*).

No total, foram convidados 21 especialistas em arquitetura de *software* e da área da educação, com relativa experiência no mercado de desenvolvimento e projeto de sistemas ou atividades de ensino e pesquisa vinculado a algumas conceituadas Universidades e Institutos federais de ensino, tais como a Universidade de São Paulo (USP/São Carlos), Universidade Federal do Ceará (UFC), Universidade Federal do Mato Grosso do Sul (UFMGs), Universidade Estadual do Piauí (Uespi) e os Institutos Federais do Ceará (IFCE) e Piauí (IFPI). No entanto, apenas catorze (14) especialistas aceitaram participar da avaliação da MOA-RA, sete (7) especialistas de cada grupo.

O primeiro grupo é formado por especialistas na área da arquitetura de *software*, com ênfase em arquitetura de referência e/ou arquitetura orientada a serviços (SOA), e alguma experiência em atividades de pesquisa e projetos arquiteturais na indústria ou academia.

Os especialistas do domínio educacional são pesquisadores e profissionais da educação envolvidos em atividades de ensino, pesquisas científicas, desenvolvimento ou familiaridade no uso de ambientes de aprendizagem ubíqua e variações (*e-learning*, *m-learning* ou *context-aware learning*).

A Seção 5.3.4 mostra o perfil dos especialistas, em relação ao tempo de experiência, tipo

de atuação, áreas de domínio, etc. As informações foram levantadas nos questionários, separadas das questões da avaliação da MOA-RA.

5.3.2 Questionários e escalas

O questionário foi elaborado com base em alguns trabalhos da literatura relacionados aos temas de arquitetura de *software* (arquitetura de referência e SOA) e ambientes educacionais, propostas por Clements e Northrop (2002); Nakagawa, Flávio e Becker (2012); Marcolino (2019); Filho (2016); etc., devidamente adaptados para contexto da aprendizagem ubíqua para abordar diferentes aspectos da documentação arquitetural da MOA-RA, tais como: modelagem, requisitos, serviços educacionais e funcionalidades dos ambientes *u-learning*, entre outros.

Também foram inseridas algumas questões para identificar o perfil dos especialistas, abordando informações sobre a titulação dos especialistas, área de concentração, cargo atual, local de trabalho, tempo e tipo de experiência com modalidades de ensino, etc. Estas informações foram analisadas e permitiram conhecer o perfil dos avaliadores, em relação aos itens e domínio de aplicação da arquitetura avaliada.

Ao final foram definidas 34 questões de múltipla-escolha para avaliar os aspectos da MOA-RA, além das questões sobre o perfil dos especialistas (10 questões). Sobre as questões que avaliaram a arquitetura, foram classificadas em três grupos de aspectos abordados:

- Aspectos gerais (8 questões): aborda os aspectos de caráter geral da arquitetura e comuns aos dois tipos de especialistas, relacionados à compreensão da proposta, clareza das descrições, objetivo da arquitetura, motivações para o seu desenvolvimento, etc;
- Aspectos arquiteturais (15 questões): trata dos aspectos mais específicos da área de arquitetura de software, direcionado para arquitetura de referência e SOA, abordando questões da modelagem, requisitos, módulos, relacionamentos, entre outros; e
- Aspectos do domínio educacional (11 questões): relacionado às questões do domínio de aplicação da arquitetura como os tipos de serviços educacionais incorporados na arquitetura, adequação a padrões educacionais, funcionalidades *u-learning*, entre outros.

A partir desta classificação foram estruturados os questionários e a documentação avaliada por cada grupo de especialistas, conforme mostra a Tabela 11. Os questionários são compostos por dois grupos de questões, sendo um grupo de questões específicas para cada um dos questionários (Aspectos arquiteturais e Aspectos do domínio educacional) e outro grupo de questões comuns aos dois questionários (Aspectos gerais).

Os questionários foram respondidos com base num conjunto de informações importantes, sintetizadas da documentação arquitetural da MOA-RA, sendo elas: informações gerais; lista de requisitos; visões arquiteturais, composta por quatro tipos diferentes de visão: (i) visão geral,

Tabela 11 – Materiais da avaliação com especialistas

Especialistas	Grupo de questões	Documentação arquitetural
Domínio educacional	Aspectos gerais Aspectos do domínio educacional	Informações gerais Lista de requisitos (GEE e ULE) Visão geral
Arquiteto de <i>software</i>	Aspectos gerais Aspectos arquiteturais	Informações gerais Lista de requisitos completa Visões arquiteturais (completa)

Fonte: Elaborada pelo autor.

(ii) visão de módulos, (iii) visão em tempo de execução e (iv) visão de implantação. Todas as visões são acompanhadas por descrições arquiteturais que ajudam a melhorar a compreensão dos aspectos importantes presentes em cada visão.

A documentação avaliada pelos arquitetos contempla todo o conteúdo referente às informações gerais e lista de requisitos, e contém todas as visões da MOA-RA. Porém, as descrições foram sintetizadas, sem deixar de abordar os aspectos mais importantes da arquitetura, devido ao grande volume de informações presentes na documentação completa da MOA-RA.

Já a documentação avaliada pelos especialistas do domínio contém as informações gerais, parte da lista de requisitos (ambientes educacionais genéricos e ambientes *u-learning*) e apenas a visão geral com suas descrições, uma vez que o restante da documentação trata das questões mais específicas da arquitetura que foge do escopo dos especialistas do domínio.

Os questionários completos, com as documentações avaliadas por cada grupo de especialistas, podem ser consultados no Apêndice A (arquitetos) e B (especialistas do domínio).

As questões têm cinco opções de resposta, baseada na escala *Likert* (LIKERT, 1932), e um campo de texto para críticas e sugestões (opcional). Cada questão apresenta uma afirmação positiva sobre determinado item da arquitetura, onde o avaliador deve responder selecionando a opção que melhor representa o nível de concordância/discordância com essa afirmação, na escala de 1 a 5 pontos.

Essa escala é bastante utilizada em pesquisas de mercado, servindo como instrumento para medir o nível de satisfação ou insatisfação do usuário em relação a um produto, serviço, atendimento, entre outros. Trata-se de uma escala simples, fácil de ser utilizada e consiste em atribuir um valor para cada opção de resposta, onde os pontos extremos (1 e 5) representam as pontuações mínimas e máximas da avaliação, e a opção central (3) é considerada ponto neutro, ou seja, não é uma pontuação positiva, nem negativa para o item avaliado.

De acordo com a Tabela 12, a escala foi utilizada para mensurar as respostas, em termos de avaliação quantitativa (opção de resposta) e qualitativa (escala de resultado).

A opção de resposta com a menor pontuação é a resposta “Discordo totalmente” (1

Tabela 12 – Escala das respostas e resultados da avaliação da MOA-RA

Escala	Opção de resposta	Escala de resultado
1	Discordo totalmente	Péssimo
2	Discordo	Ruim
3	Nem concordo, nem discordo	Regular/Neutro/Indiferente
4	Concordo	Bom
5	Concordo totalmente	Ótimo

Fonte: Elaborada pelo autor.

ponto), que é o pior resultado da avaliação e mensura o nível de qualidade “Péssimo”, pois o avaliador não concordou com nenhum quesito a favor do item abordado na questão. A opção “Discordo” também é uma resposta negativa que representa o segundo pior resultado de qualidade (Ruim), com valor de 2 pontos, pois o avaliador discorda parcialmente com a questão. A escala 3 é a resposta neutra (Nem concordo, nem discordo) e representa qualidade “regular” (nem boa, nem ruim). Já as duas opções positivas para os resultados são “Concordo” (4 pontos) e “Concordo totalmente” (5 pontos), referentes as escalas de qualidade “Bom” e “Ótimo”.

Como as respostas são de múltipla-escolha e cada questão foi respondida por mais de um avaliador, o resultado final de cada uma delas foi calculada pela média ponderada das respostas, utilizando a pontuação e quantidade de cada opção de resposta atribuída para cada questão. O resultado pode assumir qualquer valor (inteiro ou fracionário) dentro da escala. Para obter o resultado o primeiro passo foi levantar a quantidade de respostas de cada questão e aplicar esses dados na fórmula da média ponderada.

A seguir são apresentados os resultados da avaliação da MOA-RA, iniciando pelo levantamento do perfil dos avaliadores (Seção 5.3.3), resultados da avaliação com os arquitetos (Seção 5.3.4) e com os especialistas do domínio (Seção 5.3.5).

5.3.3 Perfil dos avaliadores

Também foram inseridas algumas questões para identificar o perfil dos especialistas, abordando informações sobre sua titulação, área de concentração, cargo atual, local de trabalho, tempo e tipo de experiência com modalidades de ensino, etc. Estas informações foram analisadas e permitiram conhecer o perfil dos avaliadores, em relação aos itens e domínio de aplicação da arquitetura avaliada.

Os Gráficos apresentados nas Figuras 22, 23 e 24 mostram os resultados do tipo de experiência dos arquitetos, dos especialistas do domínio e o tempo de experiência de todos os avaliadores da MOA-RA, respectivamente.

De acordo com a Figura 22, pode-se observar que a maioria dos arquitetos são experientes em pesquisa e desenvolvimento de projetos de ambientes educacionais (57%). Os demais

Figura 22 – Tipo de experiência com arquitetura de software (arquiteto)

Fonte: Elaborada pelo autor.

arquitetos têm experiência em apenas uma das atividades, ou seja, pesquisa na academia (29%) ou projetos na indústria (14%). Estes dados indicam que a MOA-RA foi avaliada por especialistas com experiência tanto em investigação científica quanto na prática de desenvolvimento e, na maioria dos casos, em ambas as atividades.

No caso dos especialistas do domínio educacional, a experiência está relacionada com quatro tipos de atividades utilizando ambientes educacionais: ensino, pesquisa, desenvolvimento e uso prático no cotidiano. Conforme a Figura 23, a maioria dos avaliadores (57%) são experientes em todas estas modalidades, ou seja, estão ou já estiveram envolvidos com atividades de pesquisa, desenvolvimento de aplicações e uso de ferramentas e ambientes de educacionais como recurso didático no processo de ensino/aprendizagem. Já, 14% dos especialistas têm experiência de uso, pesquisa e produção, 15% em pesquisa e utilização, e 14% apenas com o ensino das modalidades *e-learning*, *m-learning* ou *u-learning*.

Em relação ao tempo de experiência dos especialistas, a maioria tem mais de 5 anos de experiência na área. Conforme a Figura 24, 61% de todos os avaliadores têm mais de 5 anos de experiência, 31% tem de 2 a 5 anos e 8% tem menos de 2 anos de experiência com *u-learning* (1 avaliador). Estes números representam um tempo médio de 4 anos de experiência dos especialistas que participaram da avaliação da MOA-RA.

Baseado nos gráficos apresentados nas Figuras 23 e 24, os dados demonstram que o perfil dos especialistas é adequado para avaliar a MOA-RA uma vez que, em cada grupo de especialidade, a maioria tem experiência em todas as atividades relacionadas à sua área de conhecimento.

Outra característica importante é o tempo de experiência dos avaliadores em suas respectivas áreas (média de 4 anos), que pode ser indício para a garantia da confiabilidade dos resultados.

Figura 23 – Tipo de experiência com ambientes educacionais (esp. do domínio)

Fonte: Elaborada pelo autor.

Figura 24 – Tempo de experiência dos especialistas

Fonte: Elaborada pelo autor.

Os resultados das avaliações de cada grupo de especialistas são apresentados nas próximas seções, discutindo inicialmente os resultados do questionário do grupo dos arquitetos (Seção 5.3.5). Em seguida, a análise dos resultados dos especialistas do domínio educacional.

5.3.4 Resultados da avaliação do grupo dos arquitetos

A Tabela 13 mostra os dados levantados com base nas respostas dos questionários dos arquitetos, e está organizada em dois grupos de questões: “Aspectos gerais” e “Aspectos arquiteturais”. Os resultados de cada questão são apresentados em termos quantitativos das respostas de cada opção), com a média geral e os aspecto abordado em cada questão.

Neste grupo de especialistas, a MOA-RA foi avaliada por 7 arquitetos que responderam um questionário com 23 questões. No total, foram obtidas 161 respostas, divididas em duas

Tabela 13 – Resultado da avaliação com arquitetos

Nº	Concordo Total.	Concordo	Neutro	Discordo	Discordo Total.	Média	Questionário Aspectos Gerais
1	4	2	-	1	-	4,3	Aspecto geral
2	6	-	-	1	-	4,6	Aspecto geral
3	4	3	-	-	-	4,6	Aspecto geral
4	6	1	-	-	-	4,8	Aspecto geral
5	4	2	-	1	-	4,3	Aspecto geral
6	2	4	1	-	-	4,1	Aspecto geral
7	5	2	-	-	-	4,7	Aspecto geral
8	4	2	1	-	-	4,4	Aspecto geral
	35	16	2	3	0	4,5	Resultado Final
Nº	Concordo Total.	Concordo	Neutro	Discordo	Discordo Total.	Média	Questionário Asp. Arquiteturais
1	3	4	-	-	-	4,4	Aspectos conceituais
2	5	2	-	-	-	4,7	Aspectos conceituais
3	2	4	1	-	-	4,1	Asp. de modelagem
4	5	2	-	-	-	4,7	Asp. de modelagem
5	4	2	1	-	-	4,4	Asp. de modelagem
6	4	3	-	-	-	4,6	Infraestrutura
7	4	1	-	2	-	4,0	Aspectos conceituais
8	4	2	1	-	-	4,4	Infraestrutura
9	4	3	-	-	-	4,6	Infraestrutura
10	4	3	-	-	-	4,6	Aspectos SOA
11	4	2	1	-	-	4,4	Aspectos SOA
12	5	1	1	-	-	4,6	Aspectos SOA
13	4	3	-	-	-	4,6	Aspectos SOA
14	5	2	-	-	-	4,7	Aspectos SOA
15	4	3	-	-	-	4,6	Aspectos SOA
	61	37	5	2	0	4,5	Resultado Final

Fonte: Elaborada pelo autor.

categorias de questões: Aspectos gerais (8 questões = 56 respostas) e Aspectos arquiteturais (15 questões = 105 respostas).

Pode-se observar (Tabela 13) que nenhuma questão teve resposta “Discordo totalmente”, que é a pior opção de resposta da escala (menor valor), e que a maioria absoluta das respostas do questionário foi “Concordo totalmente” (melhor opção da escala, em termos de pontuação). Este resultado vale também para os dois grupos de questões desta avaliação: Aspectos gerais (35 respostas) e Aspectos arquiteturais (61 respostas).

Analisando especificamente os resultados do grupo de “Aspectos gerais” (Tabela 13), que recebeu 56 respostas, no total, verifica-se que três respostas (questões 1, 2 e 5) foram negativas “Discordo” e outras duas (questões 6 e 8) foram do tipo “Nem concordo, nem discordo”. As demais respostas foram todas positivas para as questões deste grupo, pois contribuem diretamente

para melhorar os resultados dos itens avaliados. Os resultados positivos foram os seguintes: “Concordo totalmente” (35 respostas) e “Concordo” (16 respostas).

A Figura 25 mostra o resultado quantitativo da avaliação dos “Aspectos gerais” da MOA-RA, onde se vê o predomínio das respostas favoráveis à arquitetura e o bom resultado obtido.

Figura 25 – Resultado quantitativo (“Aspectos gerais” – arquitetos)

Fonte: Elaborada pelo autor.

Sobre as questões (1, 2 e 5), relacionadas aos aspectos de motivação, objetivo e atendimento aos requisitos, abordados de forma clara na documentação da MOA-RA, foram as únicas questões deste grupo que receberam respostas negativas (Discordo) dos avaliadores. Porém, apesar disso, todas tiveram um bom resultado de pontos em suas médias gerais, com 4,6 pontos (questão 2) e 4,3 pontos (questões 1 e 5).

Já a questão 6 teve o pior resultado com 4,1 pontos na média geral. Nela, os especialistas avaliaram se a arquitetura incorpora elementos que tratam da segurança dos dados e usuários do ambiente educacional. Os avaliadores fizeram críticas para o item da questão (6) relatando que as descrições estão genéricas e focadas basicamente nos aspectos de segurança lógica (autenticação, *logs*, registro de atividades, etc). Também apresentaram sugestões para a melhoria da arquitetura e correções destas limitações, tais como fazer uma descrição mais contextualizada, com mais informações e detalhes, incorporando também questões da segurança de dados, no nível de infraestrutura. De fato, estas críticas são válidas uma vez que as descrições da MOA-RA apresentadas no material da avaliação foram sintetizadas do documento original, por conta do volume de informações que compõem a documentação na íntegra.

Por outro lado, o melhor resultado foi da questão 4 (4,8 pontos) que abordou sobre a compreensão geral da proposta da arquitetura, por meio da visão geral da MOA-RA. Este resultado pode ser justificado pelo fato da MOA-RA ter sido desenvolvida seguindo os princípios, regras de relacionamento, simbologias e conjunto de termos comuns da UML e padrões do

domínio (OASIS, W3C, etc).

A média geral do grupo foi de 4,5 pontos na avaliação dos arquitetos de *software*, que é considerado um resultado de pontuação alta, pois está no meio do intervalo das duas melhores pontuações da escala, entre “Bom” e “Ótimo”.

Na outra parte da Tabela 13 estão os resultados referentes ao grupo de questões de “Aspectos arquiteturais”, composto por 15 itens que abordam aspectos sobre arquitetura de *software* no domínio dos ambientes educacionais (genéricos e *u-learning*). Mais especificamente, as questões tratam de diferentes itens arquiteturais como requisitos, visões, componentes físicos, comunicação, segurança, estilos arquiteturais, terminologias, descrições, atributos de qualidade, reúso, entre outros.

Considerando a variedade dos itens, as questões foram classificadas, quanto ao tipo de aspecto abordado, em quatro grupos de itens: aspectos conceituais (questões 1, 2 e 7), aspectos de modelagem (questões 3, 4 e 5), aspectos de infraestrutura (questões 6, 8 e 9) e aspectos da arquitetura orientada a serviços (questões 10, 11, 12, 13, 14 e 15). Os resultados da avaliação destes grupos de itens são discutidos posteriormente.

Assim como os resultados do grupo anterior (Aspectos gerais), a maioria das respostas obtidas na avaliação dos Aspectos arquiteturais também foi “Concordo totalmente”, com 61 respostas, seguido pelas opções “Concordo” (37 respostas), “Nem concordo, nem discordo” (5) e “Discordo” (2). Com base nestes dados, foi feito o Gráfico (Figura 26) para mostrar que o grupo de “Aspectos arquiteturais” da MOA-RA também teve um bom resultado quantitativo, em relação ao número de respostas positivas dos itens avaliados.

Figura 26 – Resultado quantitativo (“Aspectos arquiteturais” - arquiteto)

Fonte: Elaborada pelo autor.

Em relação à média geral das questões de “Aspectos arquiteturais”, o resultado foi o mesmo do grupo de “Aspectos gerais”, igual a 4,5 pontos. Analisando os resultados individuais, as questões (2, 4 e 14) tiveram o melhor resultado do grupo com 4,7 pontos, cada. As questões

avaliaram aspectos conceituais, de modelagem e da arquitetura SOA, respectivamente.

O pior resultado foi da questão 7 com a média de 4 pontos, outra questão relacionada aos aspectos conceituais da MOA-RA e a única com duas respostas negativas (Discordo). Mesmo assim, o resultado ainda é considerado “Bom”, pois as demais respostas da questão foram todas positivas com 4 respostas “Concordo totalmente” e uma resposta “Concordo”.

Além do mais, os avaliadores ainda contribuíram para a melhoria da MOA-RA com algumas sugestões para itens variados da arquitetura, tais como: inserir mais informações na documentação, estudo de casos, cenários de aplicação, exemplos práticos, novas abordagens (REST e HTTP), experimentos de instanciação e melhoria da qualidade das imagens. Todas foram analisadas, algumas sugestões foram atendidas e outras foram apenas argumentadas.

Considerando algumas destas propostas, foram incluídas na documentação abordagens sobre o estilo arquitetural REST e o protocolo de comunicação para sistemas distribuídos HTTP, mais detalhes nas descrições de alguns itens da arquitetura (segurança e elementos transversais) e imagens com melhor qualidade gráfica.

As justificativas para as sugestões que não foram atendidas, e que consiste basicamente na inclusão de mais informações na documentação da MOA-RA, é a mesma justificativa apresentada para as críticas da questão 6 (Aspectos gerais). Algumas informações não foram incluídas integralmente na documentação avaliada e a maioria das descrições foi sintetizada da documentação original para diminuir o volume de informações do documento submetido à avaliação.

Por fim, as Figuras 27 e 28 apresentam os resultados finais (quantitativo e qualitativo) da avaliação da MOA-RA, com os especialistas arquitetos. A Figura 27 apresenta o resultado quantitativo das respostas do questionário dos arquitetos, onde fica evidente a superioridade do número de respostas “Concordo totalmente” (60%), o que representa quase 2/3 do número total de respostas da avaliação. Adicionalmente, considerando também os 33% de respostas da segunda melhor opção (Concordo), pode-se concluir que a avaliação da MOA-RA, junto aos arquitetos, teve um resultado quantitativo muito bom, uma vez que as duas melhores opções, juntas, representam 93% do resultado dos arquitetos.

Sobre os resultados qualitativos, a Figuras 28 mostra a média geral das questões dos grupos de aspectos abordados. É possível observar que todos os grupos tiveram bons resultados na avaliação de qualidade dos arquitetos, alcançando índices melhores que “Bom”.

O grupo de “Aspectos SOA” (questões 10 a 15) teve o melhor resultado com 4,6 (Figura 28) pontos e não recebeu nenhuma crítica ou sugestão dos avaliadores, demonstrando que os itens da MOA-RA atendem com qualidade os requisitos que tratam destes aspectos.

A menor média foi a do grupo de “Aspectos conceituais” (questões 1, 2 e 7) com 4,3 pontos, que ainda é um bom resultado na escala. Estas questões já foram discutidas e os resultados foram justificados, anteriormente.

Figura 27 – Resultado avaliação quantitativa (Arquitetos)

Fonte: Elaborada pelo autor.

Figura 28 – Resultado Avaliação com Arquitetos (grupo de aspectos)

Fonte: Elaborada pelo autor.

Vale ressaltar que só foram discutidos os resultados dos “Aspectos gerais” avaliados pelos arquitetos. A seguir, os resultados da avaliação com os especialistas do domínio educacional são discutidos e comparados com os dos arquitetos.

5.3.5 Resultados da avaliação do grupo dos especialistas do domínio educacional

Em relação à avaliação dos aspectos educacionais, a Tabela 14 apresenta os resultados da avaliação com os especialistas educacionais.

De acordo com a Tabela 14, pode-se observar que os resultados do grupo de “Aspectos gerais” não tiveram nenhuma resposta negativa (“Discordo” ou “Discordo totalmente”), e apenas cinco respostas foram “Neutras” para os itens avaliados neste grupo. As demais respostas foram

Tabela 14 – Resultado da avaliação com especialistas do domínio educacional

Nº	Concordo Total.	Concordo	Neutro	Discordo	Discordo Total.	Média	Questionário Aspectos Gerais
1	4	3	-	-	-	4,6	Aspecto geral
2	4	3	-	-	-	4,6	Aspecto geral
3	4	3	-	-	-	4,6	Aspecto geral
4	4	2	1	-	-	4,4	Aspecto geral
5	4	3	-	-	-	4,6	Aspecto geral
6	4	1	2	-	-	4,3	Aspecto geral
7	4	2	1	-	-	4,4	Aspecto geral
8	4	2	1	-	-	4,4	Aspecto geral
	32	19	5	0	0	4,5	Resultado Final
Nº	Concordo Total.	Concordo	Neutro	Discordo	Discordo Total.	Média	Questionário Asp. Educacionais
1	4	3	-	-	-	4,6	Asp. Educ Genéricos
2	4	3	-	-	-	4,6	Asp. Educ Genéricos
3	4	2	1	-	-	4,4	Asp. Educ Genéricos
4	4	2	-	1	-	4,3	Asp. Educ Genéricos
5	1	5	1	-	-	4,0	Asp. Educ Genéricos
6	1	4	2	-	-	3,9	Asp. Educ Genéricos
7	4	3	-	-	-	4,6	Aspectos <i>u-learning</i>
8	3	3	1	-	-	4,3	Aspectos <i>u-learning</i>
9	4	2	1	-	-	4,4	Aspectos <i>u-learning</i>
10	3	4	-	-	-	4,4	Aspectos <i>u-learning</i>
11	3	3	1	-	-	4,3	Aspectos <i>u-learning</i>
	35	34	7	1	0	4,3	Resultado Final

Fonte: Elaborada pelo autor.

todas positivas para a arquitetura, principalmente porque a maioria das respostas de cada questão do grupo foi “Concordo totalmente” (4 por questão e 32 no geral), e a segunda maior foi a opção “Concordo” com 19 respostas, no total.

A Figura 29 mostra o gráfico mostra o comparativo dos resultados dos dois grupos de especialistas (arquitetos e especialistas do domínio), na avaliação dos itens de “Aspectos gerais” da MOA-RA.

Conforme o gráfico, é possível perceber um certo grau de similaridade entre os dois resultados. A maioria das respostas foi “Concordo totalmente”, seguido pela opção “Concordo” e “Nem concordo, nem discordo”, nesta ordem. Os números das respostas são bem parecidos, em relação à quantidade destas três opções. Nenhum avaliador respondeu “Discordo totalmente” e, basicamente, a principal diferença entre os resultados foi a resposta “Discordo” registrada apenas no grupo dos arquitetos.

Em relação aos resultados de qualidade (Tabela 14), o grupo de “Aspectos gerais”

Figura 29 – Comparativo dos Resultados “Aspectos gerais” (arquitetos e esp. domínio)

Fonte: Elaborada pelo autor.

alcançou a média geral de 4,5 pontos. Nos resultados individuais, as questões (1, 2, 3 e 5) tiveram a maior pontuação do grupo com 4,6 pontos, enquanto a questão 6 teve a menor pontuação (4,3 pontos).

Pode-se verificar algumas semelhanças em relação aos resultados da avaliação qualitativa destes aspectos (Aspectos gerais), dos dois grupos de especialistas. As duas avaliações tiveram a mesma pontuação da média geral do grupo (4,5 pontos) e o pior resultado individual foi o da mesma questão (6), porém, com valores diferentes.

Sobre o grupo de “Aspectos do domínio educacional”, a Tabela 14 mostra os resultados da avaliação de dois grupos de itens importantes para os ambientes educacionais, relacionados aos “Aspectos educacionais genéricos” e “Aspectos *u-learning*”. Os itens abordam questões sobre autoria de materiais didáticos, adaptação do conteúdo, aprendizagem *just-in-time*, controle de acesso, acesso a informação e documentos, recursos para a avaliação do aluno, autoria de atividades sensíveis ao contexto, etc.

As questões de “Aspectos educacionais genéricos” tratam dos itens relacionados aos serviços e funcionalidades genéricas e comuns à maioria dos ambientes virtuais de aprendizagem (*e-learning*), tais como serviços administrativos, gerenciamento acadêmico, autoria de conteúdos multimídia, controle de acesso e demais serviços convencionais. Este grupo é composto pelas questões (1, 2, 3...6).

As demais questões (7, 8, 9, 10 e 11) estão relacionadas aos serviços de “Aspectos *u-learning*”, referentes à sensibilidade ao contexto, adaptação ao contexto, autoria de atividades de aprendizagem móvel e *context-aware*, aprendizagem em qualquer lugar e hora, etc. Os especialistas do domínio avaliaram se a documentação da MOA-RA incorpora estes itens e o quanto concordam que ela pode apoiar o desenvolvimento de ambientes com estes aspectos.

Com base na Tabela 14, é possível observar que a maioria das respostas do grupo

(Aspectos educacionais) foram as opções “Concordo totalmente” e “Concordo”, com resultados numericamente equilibrados (35 e 34 respostas, respectivamente). Nota-se também que houve apenas uma resposta “Discordo” (questão 4) e 7 “Nem concordo nem discordo” (questões 3, 5, 6, 8, 9 e 11).

A Figura 30 mostra graficamente a taxa das respostas de cada opção, onde fica evidente que o resultado geral é formado basicamente pelas duas respostas positivas (90% das respostas), sendo 46% (Concordo totalmente) e 44% (Concordo), e apenas 1% do resultado foi com resposta negativa (Discordo). Baseado nos números apresentados pode-se concluir que o resultado do grupo foi muito bom para os itens avaliados pelos especialistas do domínio. Analisando os

Figura 30 – Resultado - “Aspectos domínio educacional” (quantitativo)

Fonte: Elaborada pelo autor.

resultados individuais de algumas questões do grupo identificamos que a questão 6 teve o pior resultado da avaliação, com a média de 3,8 pontos (Tabela 14). No entanto, apesar do baixo resultado, a maioria das respostas da questão 6 foram positivas (Concordo (4) e Concordo totalmente (1)), duas neutras (Não concordo/Discordo) e nenhuma negativa (Discordo).

A questão aborda o item de “Aspectos educacionais genéricos” relacionado à personalização do ambiente educacional, ou seja, se a arquitetura de referência permite o desenvolvimento de ambientes educacionais personalizáveis, que possam ser configurados de acordo com as preferências dos usuários.

De fato, há indícios na documentação para concordar com o enunciado da questão, pois o item (personalização) está presente nos requisitos, incorporado nas visões da MOA-RA, com seus devidos relacionamentos, e abordado nas descrições do seu módulo.

Sobre a questão 4, também relacionada aos elementos de “Aspectos educacionais genéricos”, que apresenta o seguinte enunciado “A arquitetura permite o desenvolvimento de ambientes educacionais com mecanismos para realizar avaliação do aluno, de forma automati-

zada ou online”, um avaliador respondeu “Discordo”. Porém, não fez críticas ou sugestões para justificar a resposta e ajudar na identificação e correção das falhas. Apesar de ter recebido a única resposta negativa do grupo, teve o resultado geral de 4,3 pontos, pois também recebeu 4 respostas “Concordo totalmente” e duas “Concordo”. A exemplo dos argumentos apresentados na questão 6, a documentação também incorpora elementos relacionados ao item da questão 4 (avaliação do aprendiz). Estes elementos consistem em alguns requisitos arquiteturais (GEE-4, GEE-8, GEE-11 e ULE-8), módulos, serviços e relacionamentos incorporados na Visão geral “estendida” da MOA-RA e suas descrições.

O resultado qualitativo do grupo de “Aspectos do domínio educacional” da MOA-RA foi de 4,3 pontos na média geral das questões, resultado com valor acima da escala “Bom”. A Figura 31 mostra os resultados de todos os grupos de questões avaliadas pelos especialistas do domínio: “Aspectos gerais”, “Aspectos educacionais genéricos” e “Aspectos *u-learning*”.

Figura 31 – Resultado da avaliação dos especialistas do domínio (qualitativo)

Fonte: Elaborada pelo autor.

Pelo gráfico (31) é possível observar que todos os grupos de itens avaliados pelos especialistas do domínio educacional tiveram bons resultados. Todos alcançaram média acima de 4 pontos: “Aspectos educacionais genéricos” (4,3 pontos); “Aspectos *u-learning*” (4,4 pontos); e “Aspectos gerais” (4,5 pontos). A partir dos resultados dos 3 grupos de aspectos, foi obtido o resultado geral da avaliação com os especialistas do domínio, que foi igual a 4,4 pontos.

A avaliação com os especialistas foi importante para o processo de especificação da MOA-RA, pois os resultados indicam que ela atende aos principais requisitos do domínio da aprendizagem ubíqua e SOA, incorporando um conjunto de elementos básicos e necessários para que a arquitetura de referência possa apoiar o desenvolvimento de ambientes educacionais *u-learning*. Os resultados também apresentam indícios que caracterizam um bom nível de qualidade da MOA-RA, relacionado aos diferentes aspectos avaliados na arquitetura.

5.3.6 Riscos e ameaças à validade da avaliação

Deve-se levar em consideração que a avaliação da arquitetura também está sujeita a falhas que podem ameaçar a validade da arquitetura. Por isso é importante identificar as possíveis ameaças e seus fatores de riscos para os resultados da avaliação.

Na avaliação da MOA-RA (com os especialistas) foram considerados os seguintes riscos à validade da arquitetura:

- Número de avaliadores: o número de especialistas que participaram da avaliação da arquitetura foi pequeno para permitir uma análise estatística que possa garantir, cientificamente, a precisão dos resultados. Este risco ocorreu, principalmente, pelo ainda reduzido número de especialistas do domínio *u-learning* (campo ainda emergente) e também pela falta de disponibilidade no período da avaliação, alegada por alguns especialistas convidados. Porém, o número de avaliadores também não foi insignificante, a ponto de contestar a confiabilidade dos resultados (14 avaliadores);
- Documentação avaliada: a documentação da arquitetura que foi avaliada pelos especialistas foi sintetizada da documentação completa da MOA-RA. Algumas informações não estavam presentes no material avaliado (exemplos práticos de serviços, cenários de aplicação da arquitetura, descrições mais detalhadas, etc.), representando um risco que pode ameaçar o entendimento claro de alguns itens da arquitetura. A documentação foi sintetizada para evitar uma avaliação cansativa e demorada para os especialistas, devido à quantidade de informações que seriam avaliadas no documento completo. No entanto, o material avaliado contempla as informações importantes e necessárias para a instanciación de arquiteturas concretas, e passou por uma avaliação piloto que ajudou a identificar as falhas e melhorar o conteúdo do material;
- Aspectos avaliados: as questões não abordaram todos os aspectos das diferentes áreas relacionadas a MOA-RA, como arquitetura de referência, SOA, *u-learning* (*m-learning* e *context-aware*) e ambientes de autoria. Esta característica pode ser um risco para os resultados, uma vez que, os especialistas não avaliaram todos os elementos de cada domínios com profundidade. Uma avaliação completa de todos os aspectos destas áreas envolve uma quantidade muito grande de questões e, conseqüentemente, mais informações a serem avaliadas. Para minimizar este risco foram feitas outras avaliações da MOA-RA para complementar esta limitação, permitindo avaliar outros aspectos que não foram avaliados no questionário dos especialistas (comparativo com RAModel e instância arquitetural).

5.4 Considerações Finais

Este capítulo apresentou os processos de avaliação da MOA-RA e as discussões dos resultados. A arquitetura foi avaliada de duas formas, utilizando métodos e ferramentas distintas

(inspeção e avaliação com especialistas).

Na avaliação por inspeção foi feito um comparativo com o modelo RAModel (Seção 5.2), utilizando um *checklist* para verificar se a MOA-RA incorpora os elementos importantes e necessários para qualquer arquitetura de referência, definidos no modelo RAModel. Os resultados foram bem positivos para a MOA-RA, uma vez que, o *checklist* ajudou a encontrar quase todos os elementos do modelo, presentes na MOA-RA. Estes elementos são classificados em 4 grupos (domínio, aplicação, infraestrutura e elementos transversais) que cobrem muitos aspectos diferentes do escopo da arquitetura de referência. A avaliação também contribuiu para a melhoria da MOA-RA, identificando os problemas a serem corrigidos.

Na segunda avaliação foi aplicado um questionário junto a especialistas do domínio educacional e arquitetos de software, com experiência nas áreas do domínio da MOA-RA (SOA e *u-learning*). Este tipo de avaliação é importante porque os avaliadores são especialistas envolvidos em práticas de pesquisa e desenvolvimento, na academia e na indústria de software. Os questionários abordaram conjuntos de aspectos relevantes no contexto dos especialistas como aspectos arquiteturais (modelagem gráfica, modelo conceitual, descrições, conformidade a padrões, aspectos SOA, etc.) e aspectos educacionais (aspectos *u-learning*, serviços educacionais, *compliance* do sistema, etc.). O resultado desta avaliação também foi satisfatório e indica que a MOA-RA alcançou um bom nível de qualidade, segundo os especialistas, de acordo com os parâmetros da escala utilizada na avaliação.

As duas avaliações foram importantes para identificar os pontos positivos e negativos da arquitetura, nos diferentes itens e domínios avaliados, e assim, ajudaram a melhorar a arquitetura corrigindo as falhas encontradas. Ambas tiveram resultados muito favoráveis para a validade da MOA-RA, que ficaram evidentes pela superioridade numérica dos resultados positivos da arquitetura, em relação aos números negativos de cada avaliação.

No entanto, é importante ressaltar que não se pode generalizar os resultados da avaliação com os especialistas, uma vez que eles não foram analisados estatisticamente devido a quantidade de avaliadores. Mas há indícios que permitem interpretar os resultados como evidências de que os objetivos da MOA-RA foram alcançados e que ela incorpora os elementos necessários para apoiar a instanciação de arquiteturas concretas e o desenvolvimento de ambientes educacionais *u-learning*,

Por fim, para complementar o processo da avaliação da MOA-RA foram realizados dois experimentos baseados em estudos de caso, utilizando cenários de aplicação para verificar a viabilidade de uso prático da arquitetura de referência, em relação à instanciação de arquiteturas concretas e implementação de protótipos de ambientes *u-learning* orientados a serviços. Os estudos de caso da MOA-RA são apresentados no próximo capítulo.

ESTUDO DE CASO DA MOA-RA

6.1 Considerações Iniciais

As avaliações apresentadas no capítulo anterior (inspeção por *checklist* e questionário com especialistas) foram importantes para verificar os aspectos de qualidade da MOA-RA e contribuíram para a sua melhoria, identificando os problemas da arquitetura que foram corrigidos posteriormente. No entanto, essas avaliações são basicamente focadas nas questões “teóricas” da arquitetura, verificando a presença dos elementos arquiteturais, inconsistências, aspectos de modelagem, clareza, entre outros.

Arquiteturas de referência (AR) devem apoiar o desenvolvimento de famílias de *software* do mesmo domínio e podem ser instanciadas total ou parcialmente, em diferentes cenários de aplicação. Dessa forma, avaliações práticas também são importantes e necessárias para o estabelecimento de qualquer arquitetura de referência, pois permitem verificar na prática a sua aplicabilidade de uso em cenários variados.

Na engenharia de *software*, o uso de cenários é uma técnica bastante conhecida e utilizada para facilitar na identificação dos requisitos funcionais do sistema, pois descreve situações específicas de interações do usuário com o sistema e internas do próprio sistema, por meio de uma narrativa sobre a operacionalização do sistema em alto nível de abstração (FOWLER, 2005). Diante do exposto, para complementar o processo da avaliação da MOA-RA e ilustrar sua utilização prática em casos específicos, em relação à instanciação de arquiteturas concretas e implementação de protótipos de ambientes de aprendizagem móvel e ubíqua, foram feitos dois estudos de caso para verificar a viabilidade de aplicação prática da MOA-RA.

No primeiro estudo de caso, foi utilizado o protótipo de um ambiente *u-learning* construído a partir da instanciação parcial da MOA-RA para um cenário de aplicação específico. Já no segundo estudo, a arquitetura foi instanciada totalmente para implementar outro ambiente *u-learning*, aplicado em outro cenário de uso. Com isso, os estudos permitiram observar dois

aspectos importantes da MOA-RA, referentes às questões de instanciação (parcial e total) e reúso em diferentes cenários de aprendizagem.

Este capítulo apresenta os estudos de caso da MOA-RA, discutindo os aspectos envolvidos e necessários para realizar os estudos de caso. O restante do capítulo está organizado da seguinte maneira. A Seção 6.2 apresenta os cenários de aplicação utilizados para a instanciação de arquiteturas concretas, a partir da MOA-RA. A Seção 6.3 discute os aspectos relacionados à implementação dos protótipos de ambientes *u-learning*, aplicados em cada cenário. Os estudos de caso são apresentados na Seção 6.4, junto com os resultados e suas discussões. A Seção 6.5 apresenta as considerações finais do capítulo.

6.2 Cenários de Aplicação da MOA-RA

Conforme descrito anteriormente, um cenário de aplicação, no contexto da engenharia de *software*, consiste em uma narrativa completa de uma determinada situação para demonstrar o uso prático do sistema na execução de uma atividade específica (PERNAS, 2012).

No domínio da *u-learning*, um cenário pode representar uma prática de ensino e aprendizagem utilizando ambientes e tecnologias ubíquas, integradas entre si e com o ambiente educacional, tais como sensores, dispositivos móveis, GPS, etc. Essa variedade de tecnologias que podem ser utilizadas nos ambientes *u-learning*, de maneira independente ou composta de diferentes formas, oferece novas opções de ambientes educacionais, em relação às plataformas de *hardware* e *software*, permitindo novas possibilidades de cenários de aplicação.

Por isso, é importante que uma arquitetura de referência *u-learning* possa lidar com a diversidade de cenários de aplicação, permitindo a instanciação de arquiteturas concretas e ambientes educacionais heterogêneos de plataforma e de tecnologias integradas.

Nesta Seção, são apresentados os dois cenários de aplicação utilizados nos estudos de caso, representando diferentes práticas educativas e ambientes *u-learning* instanciados da MOA-RA para cada cenário.

O primeiro descreve um cenário de aprendizagem de matemática, por meio de um jogo *m-learning* sensível ao contexto (Seção 6.2.1). O segundo cenário (Seção 6.2.2) ilustra uma situação de ensino-aprendizagem de botânica, relacionada à autoria de uma atividade sensível ao contexto e execução dessa atividade em uma prática de campo, utilizando três aplicações distintas.

6.2.1 Cenário 1

Antes de mostrar o cenário propriamente dito, é apresentada uma breve contextualização de alguns aspectos importantes sobre o cenário. Entre eles, deve-se considerar que se trata de um

cenário de instanciação parcial, pois apenas parte dos requisitos da MOA-RA são necessários para desenvolver um ambiente de aprendizagem aplicado neste cenário.

O cenário-1 é um cenário de aprendizagem referente a uma prática educativa que utiliza um jogo educacional para reforçar a aprendizagem do aluno sobre o conteúdo do plano cartesiano. O jogo consiste em um aplicativo móvel, sensível ao contexto e orientado a serviços, e foi adaptado do clássico jogo de tabuleiro “*Battleship*” (batalha naval), cujo objetivo é descobrir a localização do navio adversário posicionado em um ponto da matriz de linhas e colunas do tabuleiro.

Segundo Silva (2017), o jogo batalha naval foi criado por soldados russos ao longo da Primeira Guerra Mundial. Ele é considerado um jogo de tabuleiro caracterizado pela necessidade de uso de estratégias para ser jogado e de classificação indicativa livre, no qual dois adversários tentam adivinhar a localização do navio inimigo que está posicionado em um ponto específico do tabuleiro.

O tabuleiro é representado por uma matriz $N \times M$ (N (linhas) e M (colunas)), onde cada linha e cada coluna podem ser identificadas por letras e/ou números, e a posição do navio consiste na combinação dos respectivos identificadores (linha e coluna), na qual ele se encontra. Geralmente, os tabuleiros eram desenhados em papel, papelão ou tábuas de madeira plana e os navios eram pequenas peças de plástico ou material similar.

A Figura 32 ilustra uma cena de batalha do jogo mostrando os navios devidamente posicionados em pontos específicos do tabuleiro. Neste caso, os navios estão posicionados em quadrantes da matriz e os navios posicionados em quadrantes consecutivos representam um único navio de grande porte (e.g. 1D, 1E e 1F). Durante o jogo, os navios ficam ocultos no tabuleiro para que os jogadores possam descobrir a posição do navio adversário.

Figura 32 – Tabuleiro do “*Battleship*”

	A	B	C	D	E	F	G	H	I	J	K	L	M
1				🚢	🚢	🚢							
2													
3													
4							🚢						
5											🚢		
6												🚢	
7													
8				🚢	🚢	🚢	🚢						
9													
10													

Fonte: <https://passatempoaprendendo.blogspot.com/2018/01/batalha-naval.html>

Vale ressaltar que a Figura 32 é meramente ilustrativa para demonstrar o ambiente original do jogo, no qual, a matriz do tabuleiro foi adaptada para o plano cartesiano do aplicativo utilizado

no cenário. Os navios são posicionados automaticamente pelo jogo, nos pontos cartesianos calculados e projetados no tabuleiro, a partir das coordenadas da localização atual dos jogadores, obtidas pelo GPS dos seus respectivos celulares.

Considerando o contexto apresentado, foi elaborado o seguinte cenário de aplicação (Cenário-1) para o primeiro estudo de caso, relacionado à instanciização parcial da MOA-RA:

“Para reforçar a aprendizagem sobre o plano cartesiano, os alunos devem realizar uma tarefa de casa utilizando um jogo de aprendizagem móvel, adaptado do jogo “batalha naval”. Antes da atividade, o professor ensina o assunto e passa uma lista de exercício para responder na sala de aula. Em casa, cada um deve jogar 10 batalhas por dia, cada uma com um aluno diferente, durante uma semana.

Para realizar a atividade todos os alunos devem fazer o login no jogo, na hora definida pelo professor. Em seguida, selecionar o adversário da batalha. Neste momento, os dados da localização dos jogadores, obtidos pelo GPS dos celulares, são usados para construir o plano cartesiano e posicionar seus respectivos navios.

A batalha é travada por dois jogadores que devem atingir o navio adversário, lançando bombas no local onde o navio está posicionado. Para lançar uma bomba o jogador deve informar as coordenadas cartesianas do ponto onde ela vai cair. As bombas são lançadas de forma alternada entre os jogadores, que têm 5 chances para acertar o alvo.

O jogo possui 5 níveis de dificuldade, com metas diferentes a serem alcançadas, e para passar de nível o aluno deve vencer 5 batalhas consecutivas.

No primeiro nível, o jogo constrói o tabuleiro e posiciona os navios no plano cartesiano de um único quadrante, ou seja, a área do plano representada pelos eixos x e y com valores de sinal positivo. A medida que o aluno vai passando de fase, o tabuleiro aumenta o número de quadrantes e, conseqüentemente, as opções para o posicionamento dos navios.

No último nível, o aluno deve localizar a posição do adversário, calcular a distância entre suas coordenadas e converter o valor utilizando a escala real fornecida pelo professor.

As informações de cada batalha que o aluno realizou ficam registradas no jogo, para que o professor possa acompanhar o progresso dos alunos na execução da atividade. Por exemplo, nível atual do aluno, número de vitórias, estatística de acerto, quantidade de partidas, etc.

Com base nestas informações o professor pode saber se o aluno cumpriu a atividade, identificar os alunos com dificuldades no assunto, traçar estratégias personalizadas de ensino, entre outras possibilidades.”

6.2.1.1 Instanciização Parcial da Arquitetura do Cenário-1

O cenário-1 apresentou uma situação prática de ensino de matemática, utilizando um jogo educativo para auxiliar na aprendizagem do plano cartesiano. Este cenário pode contribuir

para a aprendizagem do assunto, levando em consideração as características dos jogos educativos e da aprendizagem móvel, que podem melhorar a motivação do aluno para o estudo, estimulando seu espírito competitivo, a superação dos desafios, metas e recompensas a conquistar, etc. Diante do exposto, e considerando a necessidade de verificar a aplicabilidade prática da arquitetura de referência MOA-RA, o cenário-1 foi utilizado para a instanciação parcial da arquitetura do cenário e, posteriormente, a implementação do protótipo do respectivo jogo educacional.

A Figura 33 ilustra o processo de especialização arquitetural da MOA-RA. Inicialmente, a MOA-RA foi utilizada para construir o cenário de aplicação (cenário-1), que descreve uma situação de aprendizagem específica. Em seguida, foi feito o mapeamento dos requisitos de referência da MOA-RA, para os requisitos do sistema do cenário (jogo educativo).

Figura 33 – Processo de Especialização Arquitetural

Fonte: Duarte Filho (2016)

Antes das discussões sobre o mapeamento dos requisitos arquiteturais, é importante esclarecer que o foco deste processo foi identificar os módulos e serviços instanciados parcialmente da MOA-RA, e não nos detalhes e aspectos específicos da arquitetura concreta do jogo em si, que representa a aplicação cliente da arquitetura de referência (plataforma de aprendizagem).

Considerando este foco, a Tabela 15 mostra o resultado do mapeamento dos requisitos parciais da MOA-RA, em relação às funcionalidades consumidas pela aplicação do cenário-1.

Conforme a Tabela 15, oito requisitos do jogo foram mapeados dos requisitos de referência da MOA-RA. Os requisitos do jogo (1 e 7) foram mapeados do requisito arquitetural GEE-8, relacionado ao gerenciamento das informações do usuário e de sua aprendizagem. O requisito 7 está relacionado também ao requisito ULE-4 pois, a cada mudança de nível, o jogo deve se adaptar ao contexto do usuário, alterando as configurações visuais do tabuleiro com escala, número de quadrantes do plano cartesiano e reposicionamento dos navios, baseado no novo nível de dificuldade do jogador.

O requisito GEE-11 mapeou o requisito 2 (*login*) referente à segurança e mecanismos de controle de acesso ao sistema. Já os requisitos da aplicação (3, 4 e 8) foram mapeados do requisito ULE-1, pois estão relacionados com a captura das informações do contexto de localização (3 e

Tabela 15 – Mapeamento dos requisitos arquiteturais (cenário-1)

Nº	Requisitos do Jogo	Requisitos MOA-RA
1	Cadastrar usuários	GEE-8
2	Fazer login e logout dos usuários	GEE-11
3	Identificar o contexto atual do aluno (localização)	ULE-1
4	Obter os dados da localização dos jogadores (GPS)	ULE-1
5	Posicionar navios no tabuleiro, com base na localização dos jogadores	ULE-2
6	Lançar bombas no navio adversário	ULE-2
7	Possuir diferentes níveis de dificuldade	GEE-8, ULE-4
8	Salvar dados das partidas e interações do usuário	ULE-1

Fonte: Elaborada pelo autor.

4) e aprendizagem (8).

Por fim, os requisitos 5 e 6, relacionados ao posicionamento dos navios no plano cartesiano e o lançamento das bombas, foram mapeados do requisito ULE-2, pois são funcionalidades baseadas nas informações do contexto, que utilizam os dados de localização dos jogadores (GPS) para construir o plano cartesiano, calcular e posicionar os navios nas coordenadas (x, y).

A partir dos requisitos mapeados, foram identificados os módulos arquiteturais dos serviços instanciados pelo jogo, neste cenário de aprendizagem. A Figura 34 mostra os módulos identificados por meio do mapeamento dos requisitos.

Figura 34 – Módulos da instanciação arquitetural do cenário-1

Fonte: Elaborada pelo autor.

Conforme mostra a Figura 34, os requisitos do jogo estão associados a quatro módulos de serviços educacionais, incorporados na camada de aplicação da arquitetura MOA-RA, sendo

eles: (1) **Módulo de Administração**: relacionado ao pacote de serviços de “Administração do usuário”; (2) **Módulo de Suporte**: relacionado ao pacote de serviços de “Acesso”; (3) **Módulo de Adaptação**: relacionado ao pacote de serviços de “Adaptação ao usuário”; e (4) **Módulo de Gerência do Contexto**: relacionado aos serviços de “Aquisição” e “Serviços baseados no contexto”.

A Figura 35 ilustra a instanciação parcial da MOA-RA, apresentando a arquitetura instanciada para o jogo do cenário de aprendizagem (cenário-1), em termos da estrutura arquitetural e relacionamentos entre os módulos de serviços identificados com o mapeamento dos requisitos de referência.

Figura 35 – Instanciação parcial da arquitetura do cenário-1

Fonte: Elaborada pelo autor.

Por se tratar de uma instanciação parcial, apenas parte dos requisitos e módulos da MOA-RA foram considerados no processo dessa especialização arquitetural. Essa característica representa um aspecto importante para o contexto das arquiteturas de referência, pois flexibiliza sua aplicação em diferentes cenários, considerando apenas os requisitos de seu interesse.

De acordo com a Figura 35, pode-se observar que parte dos requisitos de referência da MOA-RA, tais como os requisitos relacionados aos conceitos de autoria, comunicação, avaliação automatizada e disponibilização de materiais didáticos, por exemplo, não são de interesse para o cenário em questão (cenário-1). Por isso, os respectivos módulos da arquitetura de referência não foram instanciados na arquitetura do cenário.

Em relação aos módulos instanciados na Camada de Aplicação, relacionados aos serviços de interesse para o cenário, pode-se observar na Figura 35 os seguintes relacionamentos e interações entre eles. Por exemplo: o módulo de Suporte, responsável pelo serviço de acesso (*login/logout*), se comunica com o módulo de Administração para fazer o login do usuário (administração do usuário); O módulo de Administração se relaciona com o módulo de Gerência do Contexto para que este possa gerenciar as informações e fornecer os serviços do contexto de cada usuário; O módulo de Adaptação tem relacionamentos com os módulos de Administração e Gerência do Contexto, pois os serviços de adaptação são baseados nas informações dos contextos do usuário e físico (localização).

Vale ressaltar que a camada de Intermediação da MOA-RA não foi instanciada neste cenário, devido ao conhecimento prévio do endereço dos serviços do provedor (serviços educacionais) pelo serviço consumidor (protótipo do jogo), uma vez que ambos foram implementados juntos e estabeleceram comunicação direta entre si.

6.2.2 Cenário 2

O segundo cenário de aplicação da MOA-RA foi definido com base na instanciação completa da arquitetura de referência, para demonstrar sua utilização prática em uma atividade de aula de campo, referente ao estudo das plantas.

Neste cenário (cenário-2), foram utilizadas três aplicações distintas para ilustrar algumas funcionalidades e interações que podem ocorrer entre as diferentes aplicações clientes da arquitetura, apresentadas no Capítulo 4 (Seção 4.4), que são: (1) Plataforma de gerenciamento; (2) Plataforma de autoria; e (3) Plataforma de aprendizagem.

A proposta deste cenário é demonstrar o processo de autoria e execução de uma atividade de aprendizagem ubíqua, e o papel de cada aplicação utilizada neste processo, em relação às interações e consumo dos serviços educacionais. Essa proposta representa um dos principais cenários de aplicação da MOA-RA, cujo objetivo é apoiar o desenvolvimento de ambientes *u-learning* para autoria de atividades de aprendizagem sensíveis ao contexto.

O cenário-2 é apresentado a seguir, descrevendo uma prática educativa baseada na instância total da MOA-RA:

“O professor de biologia quer utilizar o aplicativo de aprendizagem móvel, do novo ambiente educacional que a escola adquiriu, para criar uma atividade sensível ao contexto para uma aula de campo sobre botânica. O sistema é composto por uma aplicação web e dois

aplicativos móveis, um para os professores (plataforma de autoria) e outro para os alunos (plataforma de aprendizagem).

Para começar a utilizar o sistema, o funcionário acadêmico da escola deve cadastrar os cursos, turmas, disciplinas, professores e alunos, no ambiente educacional, por meio da aplicação web (plataforma web) acessada pelo navegador do computador. Em seguida, todos os usuários recebem um e-mail com as informações para o login das aplicações do ambiente educacional.

A aula de campo é sobre o estudo das plantas e será realizada no jardim botânico da cidade, utilizando o celular dos alunos com aplicativo instalado (plataforma de aprendizagem) para executar a atividade u-learning, criada pelo professor (plataforma de autoria), da seguinte forma: O aplicativo apresenta as informações sobre as características da espécie da planta que os alunos devem identificar e localizar no jardim botânico. Ao encontrar, o aluno deve se aproximar da planta e verificar se é a planta correta. O aplicativo compara a localização do aluno com a localização da respectiva planta de estudo, para verificar se o aluno está próximo da planta em questão.

Caso positivo, a atividade mostra uma mensagem de acerto, junto com outras características dessa planta, permitindo identificar essas características na prática, estudando a planta no seu habitat natural. Na parte final da mensagem, são apresentadas as informações da próxima planta a ser procurada na atividade. Caso contrário, a mensagem informa que o aluno não está próximo da planta certa e também mostra outras características para ajudar a encontrar a planta correta.

Para criar a atividade na plataforma de autoria, o professor foi ao jardim botânico, antes do dia da aula, para obter e registrar os dados da localização de cada planta em estudo, por meio do GPS do celular. Em seguida, criou os conteúdos e mensagens de cada planta e associou com sua respectiva localização, já registrada no sistema. Por fim, a atividade criada é automaticamente disponibilizada para o aluno, na sua plataforma de aprendizagem.”

6.2.2.1 Instanciação Total da Arquitetura do Cenário-2

Para complementar a verificação sobre a utilização prática da MOA-RA, em relação ao processo de instância arquitetural, o cenário-2 foi utilizado para a instanciação total de uma arquitetura concreta e implementação do protótipo de um ambiente de aprendizagem ubíqua.

Este cenário foi escolhido por se tratar de um cenário que descreve uma das principais situações de aprendizagem proposta para a aplicação da MOA-RA, relacionada à autoria de atividades *u-learning*. Além disso, o cenário também ilustra um ambiente educacional em um contexto tecnológico heterogêneo, envolvendo aplicações de diferentes plataformas de *hardware* e *software*, que é uma característica importante na avaliação prática da MOA-RA.

Seguindo o mesmo processo da instanciação parcial do cenário-1, foi feito o mapeamento

dos requisitos arquiteturais do cenário-2, a partir da especialização completa da MOA-RA.

Vale ressaltar que não há um consenso geral sobre quais e quantos requisitos arquiteturais devem ser utilizados e também, na prática, não é necessário utilizar todos eles, isso depende do cenário de aplicação, decisão do projetista ou necessidades dos *stakeholders* (DUARTE FILHO, 2016).

A Tabela 16 mostra os requisitos de sistema identificados em cada aplicação do cenário, assim como os requisitos e módulos arquiteturais mapeados da MOA-RA.

Tabela 16 – Mapeamento dos requisitos arquiteturais (cenário-2)

Nº	Requisitos do Sistema	Requisitos MOA-RA	Módulos arquiteturais
1	Cadastrar usuários (prof. e alunos)	GEE-8	Administração
2	Cadastrar curso (curso, turma, disciplina)	GEE-9	Administração
3	Enviar senha	GEE-4	Suporte / Comunicação
4	Fazer <i>login</i> no sistema	GEE-11	Suporte / Acesso
5	Criar conteúdo	GEE-1	Autoria
6	Criar atividade sensível ao contexto	ULE-10	Autoria
7	Disponibilizar atividade	GEE-6	Autoria / Ensino
8	Obter os dados da localização (GPS)	ULE-1	Gerência do contexto
9	Apresentar atividade	ULE-6	Gerência do contexto
10	Verificar proximidade	ULE-2	Gerência do contexto
11	Mostrar conteúdo	ULE-(2, 4, 7)	Adaptação / G. Contexto

Fonte: Elaborada pelo autor.

Conforme a Tabela 16, os requisitos foram mapeados e todos os módulos educacionais, da Camada de Aplicação da MOA-RA, foram instanciados neste cenário. Por exemplo, o requisito de sistema “Enviar senha” (requisito 3) foi mapeado do requisito arquitetural GEE-4 “Permitir comunicação síncrona e assíncrona”, requisito do grupo dos ambientes educacionais genéricos relacionado ao conceito de Comunicação. Neste caso, a plataforma *web* utilizou o *e-mail* para estabelecer comunicação individual com cada usuário (envio de senha).

Os requisitos 5 e 6 são relacionados com os serviços do módulo de “Autoria”, mapeados dos requisitos arquiteturais GEE-1 (autoria de conteúdos educacionais) e ULE-10 (autoria de atividades sensíveis ao contexto). O requisito “Disponibilizar atividade” (requisito 7) identificou dois módulos arquiteturais: **Módulo de autoria**, que disponibiliza os materiais recém-criados para o **Módulo de ensino**, cujo papel é disponibilizar o acesso aos materiais, para os alunos apropriados.

Em relação ao requisito 11 (Mostrar conteúdo), que trata da funcionalidade que deve exibir o conteúdo sobre a planta que o aluno deve encontrar, foi mapeado a partir de três requisitos arquiteturais: (i) ULE-2: “Gerenciar informações do contexto”, utilizando os dados de localização do usuário e da planta para mostrar um conteúdo relacionado a ela; (ii) ULE-4: “Adaptar ao contexto”, o conteúdo que será mostrado é adaptado ao local do aluno e da planta

que está próxima, ou seja, se for a planta correta apresenta um conteúdo, se for a planta errada o conteúdo é adaptado para exibir outra mensagem; e (iii) ULE-7: “Aprendizagem *Just-in-time*”, além do conteúdo de aprendizagem, também são apresentadas informações adicionais para ajudar o aluno a encontrar a planta certa, baseado na sua localização.

A Figura 36 ilustra o processo da especialização arquitetural do cenário-2, apresentando o mapeamento dos requisitos e módulos arquiteturais mapeados a partir da MOA-RA, na instanciação total da arquitetura do cenário.

Na próxima seção (Seção 6.3), são apresentados os protótipos dos ambientes educacionais dos dois cenários apresentados nesta seção, discutindo os aspectos da implementação dos protótipos e dos serviços consumidos em cada cenário de aprendizagem.

6.3 Implementação dos protótipos educacionais dos cenários da MOA-RA

Para complementar a avaliação prática da arquitetura de referência e encontrar indícios da viabilidade de utilização da MOA-RA, em situações reais de aprendizagem, foram implementados protótipos dos ambientes educacionais dos dois cenários de aplicação apresentados anteriormente, a partir da instanciação das suas respectivas arquiteturas concretas.

Os protótipos foram construídos junto com os serviços *web*, com objetivo de verificar questões da implementação e consumo das funcionalidades fornecidas pelos serviços educacionais genéricos e *u-learning*, propostos na arquitetura MOA-RA.

A Seção 6.3.1 apresenta as discussões sobre a implementação do protótipo do jogo educativo do cenário-1. Em seguida (Seção 6.3.2), são discutidos os aspectos sobre o desenvolvimento do ambiente educacional do segundo cenário, composto pelas três aplicações clientes da arquitetura.

6.3.1 Protótipo do jogo Batalha Naval (cenário-1)

O protótipo do jogo educativo do primeiro cenário foi implementado com base na arquitetura concreta, obtida pela instância parcial da MOA-RA. Além disso, o processo de desenvolvimento do protótipo foi baseado nos requisitos, diretrizes e organização estrutural da arquitetura de referência e boas práticas de desenvolvimento.

Parte das funcionalidades do jogo foram implementadas e disponibilizadas em serviços *web* e outras funcionalidades foram consumidas de alguns serviços disponíveis para reuso (e.g. *API Google maps*).

Para o desenvolvimento do protótipo do jogo (*front-end*) foi utilizado o *framework Ionic*, por se tratar de uma ferramenta de desenvolvimento híbrida, gratuita e de código aberto

Figura 36 – Instanciação total da arquitetura do cenário-2

Fonte: Elaborada pelo autor.

que permite utilizar tecnologias *web* aplicadas no desenvolvimento de aplicativos móveis de diferentes plataformas operacionais (*Android* e *IoS*), possibilitando o uso de todos os recursos nativos de cada plataforma (*hardware* e *software*).

O *framework Laravel* foi utilizado na implementação do *back-end*, por ser um *framework* livre e de código aberto que utiliza a linguagem de programação PHP, uma das mais utilizadas na categoria das linguagens *web* do lado servidor.

Os serviços *web* foram construídos seguindo as boas práticas da arquitetura REST, utilizando o protocolo de comunicação HTTP e JSON (*JavaScript Object Notation*), para a representação dos dados. O banco de dados utilizado escolhido foi o MySQL, que é um banco de dados relacional, eficiente e de fácil operação.

É importante destacar que, além das funcionalidades mapeadas dos requisitos da MOA-RA, também foram implementadas outras funcionalidades para o protótipo, tais como verificação de conexão à *internet*, verificação de estado de atividade do sensor GPS e de permissões concedidas, atualização de coordenadas, verificação de tempo restante, quantidade de bombas restantes, verificação de nível, contador de acertos, erros, vitórias e derrotas, entre outras. No entanto, o foco aqui são as funcionalidades dos serviços educacionais consumidos pelo aplicativo do jogo, mais especificamente, dos serviços instanciados da MOA-RA.

A Figura 37 apresenta o protótipo do jogo “Batalha naval”, mostrando a tela inicial do jogo (Figura 37 (a)), com as duas opções disponíveis para o usuário: (i) *Login*: para usuários já cadastrados no jogo (Figura 37 (b)); e (ii) *Registre-se*: para cadastrar o usuário no jogo (Figura 37 (c)).

Figura 37 – Protótipo do jogo Batalha Naval: Telas Gerais

Fonte: Elaborada pelo autor.

No primeiro acesso, o usuário deve fazer seu cadastro (*Registre-se*) para que possa definir seu nome de usuário (*nickname*) e senha de acesso (*password*). Além dessas informações, o usuário deve informar seu nome, *e-mail* e sexo, para concluir seu cadastro (Figura 37 (c)).

Em seguida, seus dados são enviados e cadastrados no servidor *web* do jogo, por meio do serviço de “Administração do usuário”. Após o registro, o usuário deve fazer o *login* no aplicativo, com o *nickname* e *password* cadastrados na sua conta (serviço de “Acesso” do módulo

de Suporte).

Uma vez logado, a tela principal do jogo é aberta (Figura 38 (a)), contendo algumas informações sobre o jogador e três botões com as seguintes opções: “Batalhar”, “Revanches” e “Perfil”. O botão “Perfil” abre uma tela contendo um conjunto de informações detalhadas sobre o perfil do jogador, em relação ao seu progresso no jogo, número de vitórias, partidas, derrotas, entre outras (Figura 38 (b)).

Os botões “Batalhar” e “Revanche” apresentam as informações dos adversários disponíveis para o usuário selecionar para uma batalha (nova ou revanche). Cada botão abre uma tela diferente, exibindo todos os adversários disponíveis (Batalhar) ou apenas os adversários que já o venceram (Revanche). As Figuras 38 (c, d) mostram as respectivas telas desses botões.

Figura 38 – Funcionalidades do jogo (Pré-batalha)

Fonte: Elaborada pelo autor.

Após selecionar um adversário, começa o processo da construção do tabuleiro da batalha. Inicialmente, o aplicativo verifica se o GPS do dispositivo móvel dos jogadores está ativo, caso contrário, aparece uma mensagem solicitando permissão para utilizar o dispositivo de *hardware* do GPS do usuário, para criar o cenário gráfico da batalha. Em seguida, o jogo obtém e envia os dados da localização dos jogadores (latitude e longitude) para o provedor de serviços do jogo (módulos de serviços de “Administração” e “Gerência do contexto”).

É importante destacar que a construção gráfica do tabuleiro, para representar visualmente os navios posicionados nas devidas coordenadas do plano cartesiano, é feita diretamente no aplicativo do jogo, com base nos resultados fornecidos pelos serviços instanciados da arquitetura e requisitados de terceiro (*API Google Maps*).

O papel dos serviços neste processo é basicamente fornecer os dados necessários para o protótipo construir o ambiente gráfico da batalha, utilizando as coordenadas de localização

do GPS, por exemplo, converter os valores da latitude e longitude, converter as coordenadas de localização em coordenadas cartesianas (x, y), calcular as distâncias das coordenadas simples e compostas para identificar os valores dos eixos e posicionar os navios nos respectivos pontos cartesianos, etc.

A Figura 39 (a) apresenta a tela de uma batalha entre dois jogadores do mesmo nível (*level 1*). Na parte superior da tela, são exibidas as informações básicas de cada jogador (patente, *nickname*, nível e vidas (barra de progresso)), logo abaixo está o tabuleiro de batalha com os navios posicionados nas devidas coordenadas (x, y) do plano cartesiano, formado por um único quadrante e escala dos eixos (x, y) com razão igual a 1, construído a partir dos dados de localização e nível dos jogadores. Na parte inferior, a tela mostra a quantidade de bombas do jogador e o tempo restante da partida. Ao lado, estão os campos de entrada das coordenadas (x e y) e o botão para lançar as bombas no ponto das coordenadas informadas pelo jogador.

As Figuras 39 (b) e 39 (c) mostram outras telas de batalha para ilustrar os diferentes tipos de tabuleiro construídos pelo aplicativo do jogo, em relação à escala e número de quadrantes do plano cartesiano, a partir de diferentes coordenadas de localização e níveis dos adversários. Na Figura 39 (b), o jogo construiu o tabuleiro com um plano cartesiano de quatro quadrantes e eixos de escala 1, para a batalha contra um adversário de maior nível (3). Por fim, com base em outra coordenada de localização e um adversário mais experiente (5), o tabuleiro foi construído utilizando uma escala de razão decimal para representar os eixos x e y do plano cartesiano, conforme mostra a Figura 39 (c).

Figura 39 – Telas de batalha do jogo

Fonte: Elaborada pelo autor.

6.3.2 Protótipo do Ambiente de Aprendizagem Ubíqua (cenário-2)

Baseado no ambiente educacional do segundo cenário, composto por três aplicações distintas e integradas, a arquitetura concreta instanciada totalmente da MOA-RA foi utilizada para a implementação dos protótipos das aplicações que representam o ambiente educacional do cenário-2.

Foram implementados três protótipos, seguindo a mesma linha do processo de desenvolvimento do jogo, ou seja, baseado nos requisitos, boas práticas, diretrizes, etc. Os protótipos do cenário-2 são: (1) *Class Path*: aplicação *web* (plataforma de gerenciamento); (2) *App-Author*: aplicativo móvel do professor (plataforma de autoria); e (3) *App-Learn*: aplicativo móvel do aluno (plataforma de aprendizagem).

Para o desenvolvimento do *front-end* de cada protótipo foi utilizada uma ferramenta de desenvolvimento específica. Por exemplo, o *Class Path* foi implementado com a linguagem de programação *Python 3.6*, por ser uma linguagem popular, extensível e que pode usar uma variedade de bibliotecas, reduzindo o código da aplicação.

Em relação ao desenvolvimento das aplicações móveis, foi utilizado o *framework Flutter*, para a implementação do protótipo *App-Author*, e o ambiente de autoria *App Inventor*, para o desenvolvimento do *App-Learn*. O *framework Flutter* é uma ferramenta de desenvolvimento *mobile*, para as plataformas *Android* e *IoS*, gratuita e de código aberto, que permite uma fácil integração com os recursos do *Google*. Já o *App Inventor* é uma aplicação *web* gratuita, com uma baixa curva de aprendizado, que utiliza uma linguagem de programação baseado em blocos, para facilitar o desenvolvimento de aplicativos móveis, na plataforma *Android*.

Para o *back-end* foi utilizado o *framework* de desenvolvimento *web* “*Django 2.2*”, *framework* escrito na linguagem de programação *Python*, de código aberto e gratuito, que permite o desenvolvimento de aplicações *web* com alto desempenho. Os serviços *web* foram construídos seguindo as boas práticas de arquitetura REST, por meio da biblioteca “*Django REST Framework 3.10*”, com os dados sendo representados no formato JSON e comunicação HTTP. O banco de dados utilizado foi o “*PostgreSQL 9.6*”.

A Figura 40 (a) mostra a tela inicial do protótipo *Class Path*, que oferece três opções ao usuário: (1) Criar instituição, que abre uma janela para cadastrar os cursos, disciplinas, turmas, professores e alunos da instituição de ensino; (2) Registrar-se como professor, opção para professores que queiram utilizar apenas alguns serviços educacionais de forma independente, ou seja, sem a necessidade de estar vinculado a uma instituição cadastrada no ambiente; e (3) *Login*, para os usuários cadastrados acessarem o sistema.

A Figura 40 (b) mostra a tela de cadastro de curso que é exibida quando o usuário clica no botão “Criar Instituição” da página inicial do protótipo. Nela, o usuário pode criar um novo curso, na opção “Clique aqui para adicionar”, e a lista dos cursos já cadastrados é exibida para inserir as turmas, disciplinas e cadastrar os alunos matriculados nas respectivas turmas de cada

Figura 40 – Protótipo do *Class Path* (plataforma de gerenciamento)

Fonte: Elaborada pelo autor.

curso, assim como “Editar” ou “Excluir” um curso. Após cadastrar um curso, o usuário deve cadastrar os professores das disciplinas do curso, na opção “Professores” do menu à esquerda da tela.

A Figura 40 (c) apresenta a tela principal do protótipo, para o usuário logado com o perfil de professor, mostrando as funcionalidades disponíveis no ambiente para o professor gerenciar suas “Disciplinas” e permitir a autoria de atividades de aprendizagem sensíveis ao contexto. Em relação aos serviços de autoria *u-learning*, o protótipo oferece três funcionalidades: (i) “Conteúdo” – para criar os conteúdos didáticos de ensino; (ii) “Localizações” – para obter e armazenar os dados de localização que serão utilizados na atividade; e (iii) “Atividades” – para criar a atividade sensível ao contexto, utilizando os conteúdos e localizações geradas com as duas funções anteriores.

É importante dizer que essas três funcionalidades de autoria, do *Class Path*, também estão presentes no protótipo do aplicativo “*App-Author*”, possibilitando também que a autoria das atividades *u-learning* possa ser feita de forma cooperativa, entre as duas aplicações. Por exemplo, o professor pode utilizar o aplicativo móvel (*App-Author*) para registrar as coordenadas de localização de cada planta que será estudada na aula de campo (jardim botânico) e criar os

seus conteúdos no *Class Path*, utilizando o computador de casa ou da escola que proporciona em um ambiente mais tranquilo e com melhor recurso visual, em relação à resolução da tela do monitor comparado à tela do dispositivo móvel.

A Figura 40 (d) mostra a tela de autoria de conteúdo didático, do *Class Path*, para ilustrar a construção de uma atividade ubíqua, utilizando os dois protótipos. Neste exemplo, está sendo criado o conteúdo de uma planta da atividade. A tela apresenta dois campos para o “Título” e a “Descrição”, que é o texto do conteúdo em si, e um botão para salvar o conteúdo.

Utilizando o protótipo “*App-Author*”, instalado no seu dispositivo móvel, o professor vai até o jardim botânico para obter as coordenadas de localização da planta referente ao conteúdo criado. Para isso, o professor deve se aproximar da planta, fazer *login* no aplicativo e na tela principal (Figura 41 (a)) selecionar a opção “Localização”. Na tela de “Localização” (Figura 41 (b)), deve inserir um “nome” que identifique a localização, uma “descrição” (opcional) e clicar no botão “Capturar localização”, na parte inferior da tela. Por fim, clicar em “Salvar”, mas somente quando as coordenadas de latitude e longitude aparecerem na tela, conforme ilustrado na Figura 41 (b).

Após criar o conteúdo e salvar sua localização, o professor já pode criar a atividade *u-learning*, selecionando a opção “Atividades” do *App-Author*. A Figura 41 (c) mostra a tela de autoria de atividades para ilustrar o processo de construção de uma atividade de aprendizagem sensível ao contexto. O primeiro passo é definir o título da atividade e a descrição do que o aluno deve fazer, por exemplo: Título – Jardim botânico (atividade 1); Descrição – “Localize a planta com essas características: Nome científico (*Mangifera Indica*), Origem (Índia), Família (*Anacardiaceae*), Frutos (comestíveis)”.

Figura 41 – Protótipo do aplicativo “*App-Author*”

Fonte: Elaborada pelo autor.

Em seguida, o professor deve associar um conteúdo a uma localização para que a atividade mostre o conteúdo para o aluno, apenas quando ele encontrar a planta da atividade e estiver próximo da sua localização física. Para isso, ele deve clicar no botão “Localização”, que mostra uma lista com todas as localizações salvas, e seleciona a da respectiva planta. O conteúdo é selecionado da mesma forma, só que com o botão “Conteúdo”. Para concluir o processo de autoria, basta selecionar a disciplina referente ao assunto da atividade e clicar no botão “Salvar”, que armazena no repositório web e disponibiliza acesso à atividade, automaticamente, no aplicativo do aluno. Vale lembrar que o conteúdo criado no protótipo *Class Path* também aparece na lista de conteúdos da tela de autoria de atividades do *App-Author*, junto com os outros que foram criados no próprio aplicativo.

Por fim, a Figura 42 apresenta o protótipo do aplicativo *App-Learn*, que representa a instanciação da plataforma de aprendizagem da arquitetura MOA-RA e deve ser instalada exclusivamente no dispositivo móvel do aluno.

Figura 42 – Protótipo do aplicativo *App-Learn* (plataforma de aprendizagem)

Fonte: Elaborada pelo autor.

A Figura 42 (a) mostra a tela de *login* do protótipo. Após a autenticação dos dados do aluno, o aplicativo mostra a lista das disciplinas que têm alguma atividade criada na plataforma de autoria (*App-Author*). A Figura 42 (b) apresenta a tela das atividades criadas na disciplina de botânica: Jardim botânico (atividade 1) e Museu (atividade 2). No local e horário da aula de campo, o aluno seleciona a atividade correspondente (jardim botânico) para iniciar a prática de estudo. Já a Figura 42 (c) ilustra a tela de execução da atividade, exibindo a descrição do que o aluno deve fazer e dois botões, sendo, um para obter sua localização e outro para verificar a proximidade, quando o aluno achar que está próximo da planta certa.

6.4 Estudos de Caso com os protótipos da MOA-RA

Para avaliar a aplicabilidade prática dos protótipos, em relação a sua utilização e aceitação por professores e alunos, diante de situações reais de ensino e aprendizagem, foram realizados dois estudos de caso baseados nos cenários de aplicação da MOA-RA (cenário-1 e cenário-2).

É importante destacar que o objetivo desses estudos não foi avaliar a aprendizagem do aluno, ou seja, se os protótipos podem contribuir para a melhoria do conhecimento do aluno, no assunto, e sim, procurar indícios de que as aplicações obtidas a partir da instanciação da MOA-RA possam ser ferramentas úteis e com boa usabilidade, frente a um cenário real da educação.

Para realizar os estudos de caso foram feitos os seguintes procedimentos: (1) Planejamento; (2) Preparação; (3) Execução; e (4) Análise dos resultados.

A etapa de “Planejamento” corresponde à fase inicial do estudo na qual foram definidas as características gerais, objetivos, participantes, instrumentos, procedimentos, entre outros aspectos necessários para a boa execução do estudo.

No final desta etapa, os estudos de caso foram definidos com base nos resultados obtidos no planejamento, que são os seguintes:

- **Objetivo geral:** os dois estudos de caso têm o mesmo objetivo geral que é avaliar o protótipo, em termos de aplicabilidade prática em um cenário real de aprendizagem;
- **Foco:** o objetivo dos estudos está focado mais especificamente para as questões relacionadas à percepção e aceitação tecnológica do usuário, no contexto de aplicação do protótipo;
- **Participantes:** professores e alunos do ensino médio e fundamental de escolas públicas e/ou privadas, envolvidos no processo de ensino/aprendizagem do plano cartesiano (estudo 1) e estudo das plantas (estudo 2) no momento que os estudos foram realizados.
- **Procedimentos:** os dois estudos consistiram na execução de atividades práticas e preenchimento do questionário avaliativo. No estudo 1, as atividades e o questionário foram as mesmas para todos os participantes, pois o estudo foi feito apenas com alunos. No estudo 2, que envolveu professores e alunos, as atividades, protótipos e questionários foram específicos para cada perfil do participante (explicado no decorrer do capítulo);
- **Materiais e equipamentos:** os instrumentos utilizados nos dois estudos de caso foram os seguintes: protótipos dos ambientes educacionais; dispositivos móveis com acesso à internet e GPS integrado; provedor de serviço *web*; questionário *online* (*google forms*); computador pessoal (*desktop/notebook*); e material impresso (termo de autorização e instruções das atividades). Vale ressaltar que os dispositivos móveis e computadores

utilizados nos estudos são de propriedade dos professores e alunos que participaram do estudo;

- Questionário de avaliação: os protótipos foram avaliados por meio de um questionário *online*, adaptado do questionário SUS (*System Usability Scale*), proposto por [Brooke et al. \(1996\)](#), como um modelo de questionário de avaliação baseado em uma escala numérica para mensurar o nível de usabilidade e satisfação do usuário com um determinado produto de *hardware* ou *software*, bastante utilizado em pesquisas de satisfação do cliente e de uso relativamente simples.

Após o planejamento, foi iniciada a etapa de “preparação”. Nesta etapa, os estudos de caso foram apresentados aos participantes, contextualizando os aspectos gerais do estudo, instruções sobre o preenchimento do questionário e atividades a serem realizadas pelos participantes. Além disso, foi disponibilizada toda a instrumentação necessária para o estudo, instalação dos protótipos nos dispositivos dos colaboradores e realizado um treinamento operacional das funcionalidades dos protótipos.

O passo seguinte foi a “execução” dos estudos de caso, no qual os participantes realizaram as atividades definidas em cada estudo, utilizando os respectivos protótipos dos ambientes educacionais. Ao final das atividades, cada participante fez a sua avaliação individual do protótipo, respondendo a um questionário online referente à ferramenta utilizada.

A última etapa foi a “análise dos resultados” dos estudos. Para isso, foi feita a extração dos dados dos questionários, referente às respostas dos participantes, e aplicados cálculos estatísticos para obter os resultados da avaliação, que foram devidamente analisados para chegar às conclusões finais dos estudos.

A seguir, são apresentadas as discussões sobre a execução dos estudos de caso, com as atividades práticas realizadas com os diferentes protótipos e cenários de aplicação.

6.4.1 Execução dos Estudos de Caso

Os estudos foram realizados com um total de 20 colaboradores, quatro professores de ciências e dezesseis alunos (matemática e ciências) do ensino médio e fundamental de escolas públicas da cidade de Parnaíba (PI).

A quantidade relativamente pequena de participantes dos estudos se deu, em parte, pelo fato deles terem sido realizados durante o período das férias escolares (mês de janeiro). Porém, apesar desse baixo número, é importante ressaltar que a principal motivação para realizar esses dois estudos de caso foi verificar a capacidade de aplicação prática da MOA-RA para apoiar o desenvolvimento de ambientes e serviços educacionais, utilizados no contexto real da aprendizagem ubíqua.

Os estudos de caso foram executados da seguinte forma:

O primeiro estudo de caso foi feito com o jogo Batalha naval, baseado no cenário de aprendizagem (Seção 6.2.1), e foi realizado com um grupo de doze alunos de matemática de uma escola municipal da cidade de Parnaíba (PI), que haviam estudado recentemente o conteúdo do “plano cartesiano”.

Pelo fato do protótipo do jogo ser uma instanciização da plataforma de aprendizagem da arquitetura MOA-RA, que representa a aplicação cliente do aluno, professores não participaram deste experimento, apenas os alunos.

O estudo foi realizado em uma semana, na qual os participantes tiveram que executar a seguinte atividade prática com o protótipo: “Jogar uma partida com cada um dos outros participantes, em dois locais diferentes”. Mais especificamente, a atividade consiste em cada participante jogar onze partidas, cada uma com um adversário diferente, quando estiver em um determinado local (*e.g.* casa), e outra rodada em um local diferente (*e.g.* praça, casa de um parente, etc.).

Essa atividade permitiu que, ao final do estudo, cada participante jogasse 22 partidas com tabuleiros diferentes, ou seja, sem repetir nenhuma vez o mesmo posicionamento dos navios no plano cartesiano.

Já o segundo estudo foi realizado para avaliar a aplicabilidade prática dos protótipos educacionais desenvolvidos para o segundo cenário de aplicação da MOA-RA, apresentado na Seção 6.2.2 (cenário-2), por meio de uma sequência de atividades executadas por quatro professores e quatro alunos da disciplina de ciências, de uma escola pública de Parnaíba (PI).

Cada protótipo foi utilizado para a execução de uma atividade específica, seguindo a ordem pré-definida no estudo, realizado no período de suas semanas, da seguinte forma:

1) Atividade 1: o professor deve utilizar o protótipo *Class Path* para cadastrar o curso, turmas, disciplinas, professores, alunos, enviar notificações, lançar notas, criar, consultar e editar conteúdos de aprendizagem;

2) Atividade 2: o professor deve utilizar o protótipo do aplicativo *App-Author* para criar uma atividade educativa sensível ao contexto, para um estudo de campo sobre plantas;

3) Atividade 3: os alunos devem utilizar o protótipo do aplicativo *App-Learn* para executar a atividade criada pelo professor, no campo de estudo.

As duas primeiras atividades do estudo de caso foram feitas pelos quatro professores colaboradores, para que os dois protótipos desenvolvidos fossem avaliados por mais pessoas (*Class Path* e *App-Author*). Porém, apenas um professor foi responsável pelo material, produzido na atividade 2, utilizado pelos alunos na execução da atividade 3 do estudo.

O local escolhido para realizar essa última atividade (3) foi à escola municipal Dr. João S. Filho (Parnaíba - PI), que tem uma grande área aberta e toda arborizada, adequado às necessidades desse estudo.

Após a conclusão das atividades, os participantes responderam os questionários de avaliação, um questionário para cada protótipo utilizado. Quatro professores responderam dois questionários diferentes (protótipo “*Class Path*” e “*App-Author*”) e quatro alunos responderam ao questionário do *App-Learn*.

A seguir são apresentadas as discussões sobre os questionários, cálculos e análise dos resultados dos estudos de caso.

6.4.2 Análise dos Resultados

Os participantes avaliaram individualmente os respectivos protótipos que utilizaram nas atividades de cada estudo de caso, por meio de um questionário de múltipla-escolha, adaptado do questionário SUS.

O questionário SUS é composto por 10 itens (questões) com 5 opções de resposta cada. Todos os itens apresentam uma frase afirmativa sobre determinada característica do protótipo, no qual, o avaliador deve selecionar a opção de resposta que indica o seu nível de concordância ou não, com a frase do item.

As opções de resposta são ponderadas, utilizando a escala de 5 pontos, da seguinte forma: Discordo totalmente/plenamente (1 ponto); Discordo (2 pontos), Neutro/não concordo, nem discordo (3 pontos); Concordo (4 pontos); e Concordo totalmente/plenamente (5 pontos).

Os itens do questionário devem ser organizados na seguinte ordem: Itens ímpares (1, 3, 5, 7 e 9) – frases com afirmações positivas; e Itens pares (2, 4, 6, 8 e 10) – frases com afirmações negativas.

O resultado final da avaliação foi calculado seguindo o método utilizado no modelo de avaliação do SUS, no qual cada item do questionário recebe uma nota do avaliador correspondente ao valor da escala da opção de resposta escolhida no item em questão. Essa nota é utilizada para calcular o *score* de cada item, utilizando duas fórmulas diferentes.

- *Score* individual dos itens ímpares: [nota do item (avaliador)] – 1;
- *Score* individual dos itens pares: 5 – [nota do item (avaliador)].

Em seguida, o valor da soma dos scores de todos os itens do questionário é multiplicado por 2,5. O resultado dessa operação representa o total de pontos da avaliação, no intervalo de 0 a 100 pontos, e pode ser classificado da seguinte forma (BANGOR; KORTUM; MILLER, 2009):

- 0 a 60 pontos: Inaceitável;
- 60 a 70 pontos: Razoável (precisa de muitas melhorias);
- 70 a 80 pontos: Bom (precisa de pequenos ajustes);

- 80 a 90 pontos: Excelente;
- 90 a 100 pontos: Nível máximo (Melhor impossível).

Baseado no modelo de avaliação do SUS, foram calculadas as notas dos questionários utilizados nos dois estudos, e os resultados da avaliação dos protótipos foram analisados individualmente. Os questionários completos da avaliação dos quatro protótipos (jogo batalha naval, *Class Path*, *App-Author* e *App-Learn*) encontram-se respectivamente nos Apêndices C a F.

A Tabela 17 mostra os resultados das avaliações dos protótipos e, em seguida, as discussões finais da análise dos dois estudos de caso.

Tabela 17 – Resultados das avaliações dos protótipos dos estudos de caso

Estudo de caso	Protótipo	Score (SUS)	Classificação (Bangor <i>et al</i> , 2009)
Estudo 1	Jogo Batalha Naval	89,7	80 – 90 (Excelente)
Estudo 2	<i>Class Path</i> (plataforma de gerenciamento)	88,1	80 – 90 (Excelente)
	<i>App-Author</i> (plataforma de autoria)	92,2	90 – 100 (Nível máximo)
	<i>App-Learn</i> (plataforma de aprendizagem)	79,5	70 – 80 (Bom)

Fonte: Elaborada pelo autor.

A partir dos questionários respondidos pelos avaliadores, foi calculado o score SUS referente à avaliação de cada protótipo, e os resultados foram classificados nas respectivas categorias da escala de Bangor, Kortum e Miller (2009). Na Tabela 17, pode-se observar que o valor do score de todos os protótipos é superior ao valor do score mínimo e aceitável da escala, igual a 70.

A seguir, são apresentadas as discussões sobre os resultados obtidos em cada estudo de caso, como prova de conceito da MOA-RA.

6.4.2.1 Resultados do Estudo 1

Em relação ao resultado do Estudo 1, o jogo Batalha naval obteve o score de 89,7 na avaliação da usabilidade e motivação do usuário (proporcionada pelo protótipo). Este resultado corresponde à escala de nível “Excelente” do modelo de classificação adotada, sendo o segundo maior nível. Sobre os itens do questionário, a melhor avaliação foi a do item 09 e a pior foi do item 04.

Sobre a avaliação do item 09, todos os participantes responderam “Concordo plenamente” com a afirmação de que as funcionalidades do jogo estão funcionando corretamente. Isso pode ser visto como um sinal positivo da aplicabilidade prática da MOA-RA, em relação aos requisitos de comunicação, consumo de serviços e gerenciamento do contexto, uma vez que nenhum avaliador identificou problemas nas funcionalidades ligadas a tais requisitos.

Já o item 04, que apresenta uma frase negativa afirmando que o jogo não ajudou a reforçar a aprendizagem do aluno sobre o assunto (plano cartesiano), dez avaliadores responderam “Discordo” e dois “Neutro”, resultando no menor *score* (2,8) dos itens do questionário. Apesar do pior resultado, nenhum avaliador concordou com a frase negativa da questão. Os demais itens ficaram com *score* médio de 3.2, do total de 4.

6.4.2.2 Resultados Estudo 2

No segundo estudo de caso foram avaliados os três protótipos do ambiente de aprendizagem ubíqua da MOA-RA. De acordo com a Tabela 17, os protótipos foram avaliados pelos participantes, obtendo os seguintes resultados individuais (*score* SUS): *Class Path* (88,1); *App-Author* (92,2); e *App-Learn* (79,5).

O *score* do *Class Path* está na faixa de nível “Excelente” da escala, como resultado da avaliação da usabilidade e motivação do protótipo. Dos dez itens do questionário, cinco itens foram avaliados com *score* máximo, que foram os itens: 01; 03; 04; 06 e 08, do questionário deste protótipo. Nestes itens, todos os avaliadores concordaram totalmente que gostariam de utilizar o *Class Path* com maior frequência (item 03) e o *layout* das telas está bem organizado, é simples e fácil de usar (item 01). Os participantes também afirmaram que o protótipo não utiliza uma linguagem difícil de entender e se sentem confiantes e motivados com as facilidades do serviço de autoria *u-learning*.

Por outro lado, os itens 02 e 07 tiveram o menor *score* do questionário (2,75). O item 02 aborda sobre a dificuldade para encontrar as funcionalidades do protótipo, e um avaliador concordou com a abordagem. No entanto, todos os outros discordaram parcial ou totalmente da questão. No item 07, dois avaliadores foram neutros e dois concordaram que a aplicação permite repassar o conteúdo de aprendizagem, de maneira mais eficiente.

Em relação ao aplicativo *App-Author*, utilizado na atividade de autoria do conteúdo *u-learning*, foi o protótipo que teve o maior *score* SUS, entre todos os protótipos avaliados nos dois estudos, igual a 92,5 pontos. Este valor corresponde ao resultado de nível máximo da escala, no intervalo de 90 a 100 pontos. Dos dez itens do questionário do *App-Author*, apenas o item 08 recebeu a única resposta negativa da avaliação do protótipo, na qual um avaliador concorda parcialmente que não acha que o aplicativo possa motivar o professor, por facilitar o processo de autoria de atividades *u-learning*, porém, a maioria dos demais avaliadores respondeu com a opção “Discordo totalmente” para a afirmação negativa do item 08. Além disso, houve apenas uma resposta “Neutra” na avaliação do *App-Author* (item 10), a respeito das funcionalidades do aplicativo estarem bem integradas entre si.

Fora isso, as demais respostas dos questionários dos avaliadores foram todas parcialmente ou totalmente positivas para o protótipo, e cinco itens foram avaliados com *score* máximo, que são os itens 01, 02, 05, 07 e 09. Nestes itens os participantes foram unânimes, em relação aos seguintes aspectos: Simplicidade e organização estrutural do *layout* gráfico das telas do

protótipo (item 01); Facilidade para encontrar as funcionalidades (item 02) e navegar entre as telas (item 05); Capacidade de aumentar a motivação do aluno, criando materiais didáticos atraentes, interativos e com boa mobilidade (item 07); e permite que usuários com conhecimentos básicos em informática possam criar atividades de aprendizagem móvel e sensível ao contexto, sem a necessidade de conhecimentos avançados em informático ou programação (item 09).

Por fim, o aplicativo *App-Learn* foi o último protótipo avaliado no segundo estudo de caso e ficou com *score* de 79,5 pontos, a menor pontuação individual (protótipo) obtida nos dois estudos. Mesmo sendo o pior *score*, em comparação com os *scores* discutidos anteriormente, o resultado é classificado no nível “Bom” da escala, como mostra a Tabela 17.

Os itens 01, 05 e 07 tiveram os melhores resultados do questionário dos participantes, que avaliaram bem o *design* das interfaces, navegabilidade e capacidade do protótipo em apoiar a aprendizagem, apenas com as opções de respostas positivas para o aplicativo.

Dois itens do questionário tiveram uma avaliação negativa, cada. No item 02, um avaliador respondeu que teve dificuldade para encontrar as funcionalidades do aplicativo e, no item 04, outro avaliador respondeu que o *App-Learn* utiliza uma linguagem complexa que dificulta a compreensão de algumas informações, a primeira vista. Por outro lado, os demais avaliadores responderam o contrário, nestes dois itens (02 e 04).

O ponto negativo da avaliação do *App-Learn*, que resultou no pior *score* dos dois estudos de caso foi a quantidade de respostas neutras dos avaliadores, com aproximadamente 18 do total de respostas dos questionários. Seis itens receberam pelo menos uma resposta do tipo “Neutro” (03, 04, 06, 08, 09 e 10). Apesar dessa opção (Neutro) não indicar uma avaliação negativa do item, também não indica que é positiva para tal.

Para finalizar, a Figura 43 mostra o gráfico comparativo entre os protótipos dos dois estudos de caso, em relação ao *score* médio obtido nas questões de motivação, usabilidade e percepção de cada protótipo, por parte dos avaliadores.

Figura 43 – Comparativo entre os protótipos dos estudos de caso (motivação, usabilidade e percepção)

Fonte: Elaborada pelo autor.

6.5 Considerações Finais

Este capítulo abordou questões importantes para o contexto desta pesquisa, relacionadas à instanciamento arquitetural e implementação de protótipos de ambientes de aprendizagem ubíqua para cenários reais de ensino e aprendizagem, tendo como proposta complementar a avaliação da MOA-RA.

Para verificar a aplicabilidade prática da arquitetura de referência proposta, diante de diferentes contextos de uso, foram definidos dois cenários educacionais para a especialização de arquiteturas concretas, a partir da instanciamento parcial e completa da MOA-RA.

Com base na arquitetura parcialmente instanciada, foi implementado o protótipo do jogo Batalha naval, e outros três protótipos (*Class path*, *App-author* e *App-learn*) foram construídos a partir da instanciamento completa da MOA-RA. Neste processo, também foram construídos os serviços educacionais consumidos pelos protótipos de cada cenário de aplicação, baseados nos requisitos arquiteturais da MOA-RA.

Além disso, foram realizados dois estudos de caso com professores e alunos utilizando os protótipos da MOA-RA para executar atividades práticas, nos respectivos cenários de aprendizagem. Por meio destes estudos, foi possível verificar que as funcionalidades dos protótipos executaram corretamente, sem apresentar erros ou inconsistências ocasionadas pelo consumo dos serviços, de acordo com os participantes dos estudos.

Os protótipos também foram bem avaliados nas questões de usabilidade, motivação e percepção do seu potencial favorecimento ao processo do ensino e aprendizagem, fornecendo indícios da aplicabilidade prática da MOA-RA para apoiar a instanciamento arquitetural e implementação de ambientes de autoria e aprendizagem ubíqua.

CONCLUSÃO

7.1 Considerações Iniciais

Este capítulo apresenta as conclusões desta tese de doutorado, mostrando uma visão geral acerca do trabalho (Seção 7.2), principais contribuições alcançadas com o desenvolvimento da MOA-RA (Seção 7.3), dificuldades e limitações da arquitetura proposta (Seção 7.4), propostas para trabalhos futuros (Seção 7.5) e as publicações decorrentes dessa pesquisa de doutorado (Seção 7.6).

7.2 Visão Geral do trabalho

Esta tese de doutorado investigou e definiu uma arquitetura de referência para ambientes de aprendizagem ubíqua, chamada MOA-RA, com o objetivo de fornecer apoio para projetos arquiteturais de ambientes *u-learning*. Mais especificamente, a arquitetura é focada em ambientes de autoria de conteúdos e aplicações de aprendizagem sensíveis ao contexto, para dispositivos móveis.

Os potenciais benefícios da aprendizagem móvel e ubíqua para o processo de ensino-aprendizagem, aliado à complexidade do desenvolvimento de ambientes *u-learning* e a relativa carência de arquiteturas de referência para apoiar o desenvolvimento de ferramentas de autoria *u-learning*, que possam ser utilizadas para facilitar a construção de conteúdos ou aplicações sensíveis ao contexto, sem utilizar mecanismos de geração automática do código, serviram como cenários de motivação deste trabalho, que foi desenvolvido com base nas quatro etapas do processo ProSA-RA.

Na primeira etapa (RA-1) foram realizadas pesquisas na literatura que permitiram identificar e, em seguida, analisar trabalhos relevantes ao tema, dos quais foi levantado um conjunto de informações importantes para o desenvolvimento da arquitetura MOA-RA. Por meio da investiga-

ção dessas fontes de informação foi possível conhecer o estado da arte, os problemas e soluções propostas na literatura, que ajudaram a definir as características, tecnologias e funcionalidades dos ambientes educacionais que foram mapeadas para os requisitos arquiteturais, na segunda etapa do processo.

Para facilitar a etapa de análise dos requisitos arquiteturais (RA-2), as informações foram separadas por categorias que permitiram definir três grupos de requisitos da MOA-RA: (GEE) requisitos de ambientes educacionais genéricos; (ULE) requisitos de ambientes *u-learning*; e (SOA) requisitos da arquitetura SOA.

A partir dos requisitos especificados, foi iniciada a etapa RA-3, correspondente à fase do projeto da arquitetura de referência. Nesta etapa, foi elaborada a documentação arquitetural, composta pelos conceitos, descrições e visões gráficas que representam a modelagem da arquitetura. Com base nas recomendações do ProSA-RA, foram construídas quatro diferentes visões da MOA-RA, sendo elas: (1) Visão Geral; (2) Visão de Módulo; (3) Visão em Tempo de Execução; e (4) Visão de Implantação.

Por fim, seguindo a última etapa do processo (RA-4), foram feitas as avaliações arquiteturais. Como não há na literatura um modelo único e completo para avaliar arquiteturas de referência, a MOA-RA foi avaliada de três formas distintas, que foram: (1) avaliação por *checklist*, baseada nos critérios necessários para qualquer arquitetura de referência, propostos no modelo de referência RAModel (NAKAGAWA; FLÁVIO; BECKER, 2012); (2) avaliação com especialistas, por meio de questionários aplicados a dois grupos de especialistas do domínio educacional e arquitetos de software; e (3) instanciação arquitetural, como prova de conceito da aplicabilidade prática da MOA-RA, realizada por meio de duas instâncias da arquitetura de referência (parcial e total).

Além disso, ainda foram implementados quatro protótipos a partir das arquiteturas instanciadas da MOA-RA, que foram aplicados em dois estudos de caso baseados em cenários de aprendizagem reais. Apesar do número reduzido de avaliadores que participaram dos estudos, todos os protótipos tiveram bons resultados na avaliação da usabilidade, motivação e percepção do papel de cada um no processo de ensino-aprendizagem.

Considerando os resultados obtidos nas avaliações da arquitetura proposta, apresentados e discutidos no decorrer deste trabalho, foi possível levantar informações para responder às questões de pesquisa levantadas nesta tese, da seguinte forma:

- ***"Como professores sem conhecimento em programação podem construir conteúdos sensíveis ao contexto para a plataforma dos dispositivos móveis?"*** - (resposta): Professores leigos em programação poderão utilizar ferramentas de autoria, cuja construção foi facilitada pela MOA-RA, para criar de forma simples seus próprios conteúdos *u-learning*, uma vez que tais ferramentas de autoria encapsulam o código das funcionalidades necessárias para a construção dos conteúdos sensíveis ao contexto, em comandos com um nível

alto de abstração, fornecendo um ambiente com boa usabilidade para o professor;

- **Como uma ferramenta de autoria pode criar atividades u-learning para dispositivos móveis, sem utilizar mecanismos de geração automática do código da atividade criada?** - (resposta): Disponibilizando duas plataformas diferentes, que possam se comunicar por meio de serviços *web* para permitir a interoperabilidade entre as aplicações. Uma plataforma, de autoria, representa a interface do professor que propõe as atividades e outra plataforma representa o aplicativo do aluno, que recebe e executa os conteúdos das atividades sensíveis ao contexto;
- **Como garantir a persistência do conteúdo de uma atividade sensível ao contexto, alterado após sua entrega?** - (resposta): Por meio dos serviços *web* responsáveis pelo fornecimento dos conteúdos criados pela plataforma de autoria. A persistência é feita no repositório *web*, garantindo assim que os conteúdos consumidos pelas aplicações-cliente, na forma de serviços *web*, estejam com os dados íntegros e sempre atualizados.

Desta forma, considerando o conjunto da obra apresentado nesta pesquisa, por meio das discussões, análises, experimentos e resultados, pode-se concluir que há evidências que possam confirmar a hipótese levantada nesta tese, mostrando que os objetivos foram alcançados e a arquitetura MOA-RA foi validada.

7.3 Contribuições

As principais contribuições desta tese de doutorado foram as seguintes:

- Especificação de um conjunto de requisitos para projetos arquiteturais de ambientes educacionais genéricos, relacionados às características e serviços mais comuns dos ambientes virtuais de aprendizagem, assim como dos ambientes *u-learning* baseados em serviços. Os requisitos foram validados por comparação e instanciação arquitetural, e permitem o reuso em diferentes projetos do mesmo domínio;
- Levantamento e caracterização do estado da arte dos ambientes de aprendizagem ubíqua, autoria *u-learning*, arquiteturas concretas de ambientes de aprendizagem móvel e sensível ao contexto, arquiteturas de referência e SOA;
- Construção de duas arquiteturas concretas, a partir da instanciação da MOA-RA para diferentes cenários de aplicação, e implementação de quatro protótipos de ambientes educacionais (jogo batalha naval, *Class path*, *App-author* e *App-learning*);
- Documentação arquitetural composta por quatro visões da MOA-RA (visão geral, visão de módulos, visão de implantação e visão de infraestrutura) e suas respectivas descrições que

fornece um vocabulário comum para facilitar o entendimento dos aspectos relevantes relacionados ao domínio do projeto.

7.4 Dificuldades e Limitações

Durante a execução deste trabalho, foram encontradas algumas dificuldades técnicas e operacionais, geradas por determinados fatores que resultaram nas limitações da arquitetura MOA-RA. A falta de precisão dos dispositivos GPS, que consideram uma margem de erro de aproximadamente 10 metros, foi uma dificuldade encontrada nos testes funcionais dos protótipos, que pode inviabilizar a execução de atividades que devem calcular a proximidade do usuário, quando dois ou mais objetos estiverem localizados a poucos metros de distância entre si. Outra dificuldade de nível tecnológico é a dependência da Internet para executar as atividades sensíveis ao contexto, uma vez que os serviços responsáveis são executados no servidor *Web*.

Além disso, a quantidade e diversidade temática envolvida no escopo deste trabalho representou mais uma dificuldade enfrentada, pois exigiu um grande esforço e consumo de tempo para investigar a fundo as diferentes áreas abordadas nesta pesquisa, tais como: arquitetura de referência, SOA, aprendizagem móvel, *u-learning* e autoria de conteúdos educacionais sensíveis ao contexto.

Uma limitação deste trabalho está relacionada ao alto nível de abstração dos requisitos da MOA-RA. Por se tratar de uma arquitetura de referência, os requisitos foram definidos de forma abrangente para atender às diferentes propostas e necessidades relacionadas ao domínio de aplicação da arquitetura, sem descrever com maiores detalhes aspectos mais específicos para cada ambiente instanciado da arquitetura. Esse fato pode ser visto como limitação, mas simultaneamente trata-se de uma característica inerente a arquiteturas de referência, as quais devem ser flexíveis o suficiente para alcançarem maior nível de reutilização.

Outra limitação está relacionada ao foco da avaliação da aplicabilidade prática da MOA-RA que, neste momento, ficou restrito às questões do contexto da localização, utilizando apenas a tecnologia dos dispositivos GPS para a captura das informações do contexto. Tais limitações se devem, em parte, pela dificuldade relacionada ao escopo desta pesquisa, o que comprometeu a realização de uma avaliação mais completa e abrangente de todos os aspectos e áreas do domínio de aplicação da arquitetura de referência proposta.

7.5 Trabalhos Futuros

Para dar continuidade às pesquisas decorrentes desta tese de doutorado, foram definidas as seguintes atividades, a serem realizadas como trabalhos futuros:

- Elaboração e disponibilização de um catálogo de requisitos atualizado e com informações

mais detalhadas sobre cada requisito arquitetural da MOA-RA, para facilitar o entendimento e reuso por outros projetos arquiteturais da área;

- Realizar novas avaliações da arquitetura com um número maior de avaliadores, explorando os demais serviços educacionais da MOA-RA que não foram especificamente contemplados nas avaliações feitas até o momento;
- Modelar outras visões da MOA-RA, tais como a visão conceitual e visão de processos, para complementar a documentação da arquitetura de referência;
- Investigar e definir novos módulos, serviços e tecnologias que serão incorporados na arquitetura como parte do processo da evolução arquitetural da MOA-RA; e
- Instanciação de novas arquiteturas concretas para outros tipos de ambientes e cenários de aplicação, seguida da implementação dos respectivos ambientes *u-learning*.

7.6 Publicações Científicas da Pesquisa

No decorrer desta pesquisa de doutorado, foram publicados alguns artigos em eventos científicos na área de domínio da MOA-RA, no âmbito internacional, nacional e regional. Além disso, outro artigo foi submetido ao *Journal "INFORMATICS IN EDUCATION"*, e ainda está aguardando o resultado. As publicações resultantes da presente tese foram:

- LOPES, Átila Rabelo; BARBOSA, Ellen Francine; BRAGA, Rosana Teresinha Vaccare. *Service Oriented Reference Architecture for the Development of Context-aware Learning Environments*. In: 2019 IEEE Frontiers in Education Conference (FIE), 2019, Covington. 2019 IEEE Frontiers in Education Conference (FIE), 2019. v. 1. p. 1-8. DOI: 10.1109/FIE43999.2019.9028596.
- LOPES, Átila Rabelo; OLIVEIRA, Daniel Carvalho de; AGUIAR, Ruan C. de Sousa; BRAGA, Rosana Teresinha Vaccare. *Context-aware ubiquitous learning: Literature systematic mapping on ubiquitous learning environments*. In: 2017 International Symposium on Computers in Education (SIIE), 2017, Lisboa. 2017 International Symposium on Computers in Education (SIIE), 2017. v. 1. p. 1-6.
- SILVA, Lucas da C.; LOPES, Átila R.; OLIVEIRA, Daniel Carvalho de; CALAÇA, Francisco Augusto; BRAGA, Rosana Teresinha Vaccare. Um Jogo Educacional *U-Learning* no Processo de Ensino e Aprendizagem da Geometria Analítica. In: Escola Regional de Informática do Piauí, 2018, Teresina, PI, Brasil. Anais da ERI-PI. Porto Alegre - RS - Brazil: Sociedade Brasileira de Computação, 2018. v. 1. p. 1-68.
- MENDONÇA, Karoene D. S.; OLIVEIRA, João V. A. de; LOPES, Átila R.; BRAGA, Rosana Teresinha Vaccare. Ambiente de Aprendizagem Ubíqua para Auxiliar o Estudo de

Botânica em Atividades de Aula de Campo. In: XXIX Simpósio Brasileiro de Informática na Educação (*Brazilian Symposium on Computers in Education*), 2018, Fortaleza, CE, Brazil. Anais do SBIE 2018, 2018. v. 1. p. 1738-1740. DOI: 10.5753/cbie.sbie.2018.173.

REFERÊNCIAS

ABECH, M. E. **EduAdapt: um modelo de adaptação de objetos de aprendizagem com foco em dispositivos móveis**. Dissertação (Mestrado) — Universidade do Vale do Rio dos Sinos, São Leopoldo, RS., 2014. Citado na página [73](#).

AFFONSO, F. J.; NAKAGAWA, E. Y. A reference architecture based on reflection for self-adaptive software. In: IEEE. **2013 VII Brazilian Symposium on Software Components, Architectures and Reuse**. [S.l.], 2013. p. 129–138. Citado na página [66](#).

ANGELOV, S.; TRIENEKENS, J. J.; GREFFEN, P. Towards a method for the evaluation of reference architectures: Experiences from a case. In: **Proc. of the 2nd Eur. Conf. on Software Architecture**. [S.l.: s.n.], 2008. Citado nas páginas [25](#), [32](#) e [33](#).

AUGUSTIN, I.; FERREIRA, G. P.; YAMIN, A. Grade computacional como infra-estrutura para a computação pervasiva/ubíqua. In: **ERAD**. [S.l.: s.n.], 2008. Citado na página [40](#).

BANGOR, A.; KORTUM, P.; MILLER, J. Determining what individual sus scores mean: Adding an adjective rating scale. **Journal of usability studies**, v. 4, n. 3, p. 114–123, 2009. Citado nas páginas [149](#) e [150](#).

BARBOSA, D. N. F. **Um Modelo de Educação Ubíqua Orientado à Consciência do Contexto do Aprendiz**. Tese (Doutorado) — Universidade Federal do Rio Grande do Sul, 2007. Citado na página [24](#).

BASS, L.; CLEMENTS, P.; KAZMAN, R. **Software Architecture in Practice**. [S.l.]: Addison-Wesley, 2003. Citado nas páginas [25](#), [29](#), [30](#) e [31](#).

BAYER, J.; FORSTER, T.; GANESAN, D.; GIRARD, J.-F.; JOHN, I.; KNODEL, J.; KOLB, R.; MUTHIG, D. Definition of reference architectures based on existing systems. **IESE-Report**, Citeseer, 2004. Citado na página [33](#).

BRAMBILLA, M.; CABOT, J.; WIMMER, M. Model-driven software engineering in practice. **Synthesis Lectures on Software Engineering**, v. 1, n. 1, p. 175–182, 2012. Citado na página [24](#).

BRAVO, J. F. T.; ROBLES-BYKBAEV, V. E.; NORES, M. L.; ORDOÑEZ-MORALES, E. F.; BLANCO-FERNÁNDEZ, Y.; GIL-SOLLA, A. Oppia: A context-aware ubiquitous learning platform to exploit short-lived student networks for collaborative learning. In: **CSEDU (1)**. [S.l.: s.n.], 2016. p. 494–498. Citado nas páginas [24](#), [26](#), [49](#), [50](#), [67](#) e [78](#).

BROOKE, J. *et al.* Sus-a quick and dirty usability scale. usability evaluation in industry 189, 194 (1996), 4–7. **This page intentionally left blank**, 1996. Citado na página [147](#).

CASTRO, G. G.; DOMINGUEZ, E. L.; VELAZQUEZ, Y. H.; MATLA, M. R.; TOLEDO, C. E.; HERNANDEZ, S. P. Mobilearn: context-aware mobile learning system. **IEEE Latin America Transactions**, IEEE, v. 14, n. 2, p. 958–964, 2016. Citado nas páginas [70](#) e [71](#).

- CHIANG, T. H.; YANG, S. J.; HWANG, G.-J. Students' online interactive patterns in augmented reality-based inquiry activities. **Computers Education**, v. 78, p. 97–108, 2014. Citado na página 24.
- CLEMENTS, P.; NORTHROP, L. M. **Software Product Lines: Practices and Patterns**. [S.l.]: Addison-Wesley, 2002. Citado nas páginas 29, 30 e 111.
- DUARTE FILHO, N. F. **Uma contribuição ao estabelecimento de uma arquitetura de referência para ambientes de aprendizagem móvel**. Tese (Doutorado) — Universidade de São Paulo, 2016. Citado nas páginas 25, 32, 33, 45, 53, 62, 66, 67, 71, 72, 76, 79, 80, 89, 94, 98, 100, 105, 131 e 136.
- ERL, T. **Service-oriented architecture: Concepts, technology and design**. [S.l.]: Prentice Hall, 2005. Citado nas páginas 25, 26, 30, 31, 32, 35, 37 e 39.
- ESCUADERO, A.; DÍAZ, J. C. T.; GOMEZ, J.; MONTORO, G. Technology: aided authoring tools for asd coaching support. In: **Proceedings of the XIX International Conference on Human Computer Interaction**. [S.l.: s.n.], 2018. p. 1–6. Citado nas páginas 26, 51, 63, 70, 71, 78 e 79.
- FIELDING, R. T. **Architectural styles and the design of network-based software architectures**. [S.l.]: University of California, Irvine Irvine, 2000. v. 7. Citado na página 39.
- FILHO, E. d. B. M. Ufc-inventor: um ambiente para modelagem e geração de aplicações para aulas de campo ubíquas. 2016. Citado nas páginas 24, 25, 26, 40, 42, 43, 45, 61, 62, 67, 69, 71, 79, 80 e 111.
- FIORAVANTI, M. L.; NAKAGAWA, E. Y.; BARBOSA, E. F. Educar: Uma arquitetura de referência para ambientes educacionais. In: **Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)**. [S.l.: s.n.], 2010. v. 1, n. 1. Citado nas páginas 66, 71, 79, 80 e 89.
- FLÔRES, M. L. P.; TARROUCO, L. M. R.; REATEGUI, E. B. Funcionalidades da ferramenta de autoria para apoiar a construção de objetos de aprendizagem. In: **Anais do Simpósio Brasileiro de Informática na Educação (SBIE)**. [S.l.: s.n.], 2011. Citado nas páginas 24 e 44.
- FOWLER, M. **UML Essencial: Um breve guia para a linguagem-padrão de modelagem de objetos**. [S.l.]: Bookman, 2005. Citado na página 127.
- FUGITA, H. S. **MAPOS: método de análise e projeto orientado a serviços**. Tese (Doutorado) — Universidade de São Paulo, 2009. Citado na página 35.
- GALLAGHER, B. P. **Using the architecture tradeoff analysis methods to evaluate a reference architecture: a case study**. [S.l.], 2000. Citado na página 32.
- GARCÍA, C. G.; Castro, G.; Rodriguez, M. Y.; Toledo, C. B. Midgar: Generation of heterogeneous objects interconnecting applications. a domain specific language proposal for internet of things scenarios. **Computer Networks**, v. 64, p. 143–158, 2014. Citado nas páginas 24, 26, 59 e 60.
- GIEMZA, A.; BOLLEN, L.; HOPPE, H. U. Lemonade: field-trip authoring and classroom reporting for integrated mobile learning scenarios with intelligent agent support. **International Journal of Mobile learning and Organization**, v. 5, n. 1, p. 96–114, 2011. Citado na página 24.

GORTON, I. **Essential software architecture**. [S.l.]: Springer Science & Business Media, 2006. Citado nas páginas 32 e 33.

GÓMEZ, S.; ZERVAS, P.; SAMPSON, D. G.; FABREGAT, R. Context-aware adaptive and personalized mobile learning delivery supported by uolmp. **Journal of King Saud University - Computer and Information Sciences**, v. 26, n. 1, Supplement, p. 47 – 61, 2014. ISSN 1319-1578. Current Advances in Digital Learning Technologies. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S1319157813000372>>. Citado nas páginas 24 e 26.

HANSEN, M. D. **SOA using java web services**. [S.l.]: Pearson Education, 2007. Citado na página 39.

HSU, T.-Y.; CHIOU, C.-K.; TSENG, J. C.; HWANG, G.-J. Development and evaluation of an active learning support system for context-aware ubiquitous learning. **IEEE Transactions on Learning Technologies**, IEEE, v. 9, n. 1, p. 37–45, 2015. Citado nas páginas 24, 51, 52, 62, 70 e 71.

HWANG, G. J.; WU, C. H.; TSENG, J. C. R.; HUANG, I. Development of a ubiquitous learning platform based on a real-time help-seeking mechanism. **Br. J. Educ. Technol**, v. 42, n. 6, p. 992–1002, 2011. Citado nas páginas 23 e 43.

LOPES, A. R.; OLIVEIRA, D. C. de; AGUIAR, R. C. de S.; BRAGA, R. T. V. Context-aware ubiquitous learning: Literature systematic mapping on ubiquitous learning environments. In: **2017 International Symposium on Computers in Education (SIIE)**. [S.l.: s.n.], 2017. p. 1–6. Citado nas páginas 23, 42, 44, 47 e 66.

MACHADO, C. A. N. **Uma linguagem de descrição arquitetural baseada em uma**. Tese (Doutorado) — Universidade Federal do Rio Grande do Norte (Programa de Pós-Graduação em Ciência da Computação), 2015. Citado na página 25.

MACHADO, R.; SANTI, B.; PERNAS, A. M.; YAMIN, A. Ambientes educacionais ubíquos: uma proposta explorando processamento semântico para ciência de contexto. In: IEEE. **Anais do XXVIII Simpósio Brasileiro de Informática na Educação (SBIE 2017). VI Congresso Brasileiro de Informática na Educação (CBIE 2017)**. [S.l.], 2017. p. 1387–1396. Citado nas páginas 43, 44 e 45.

MACKENZIE, C. M.; LASKEY, K.; MCCABE, F.; BROWN, P. F.; METZ, R.; HAMILTON, B. A. Reference model for service oriented architecture 1.0. **OASIS standard**, v. 12, n. S 18, 2006. Citado na página 36.

MANWARING, K.; CLARKE, R. Surfing the third wave of computing: A framework for research into eobjects. **Computer Law Security Review: The International Journal of Technology Law and Practice**, v. 31, p. 586, 08 2015. Citado na página 40.

MARÇAL, E.; ANDRADE, R.; VIANA, W. Ufc-inventor: Uma ferramenta de autoria dirigida por modelos para geração de aplicações ubíquas para aulas de campo. **Revista Brasileira de Informática na Educação**, v. 27, p. 132–149, 09 2019. Citado nas páginas 42 e 43.

MARCOLINO, A. d. S. **Uma infraestrutura de apoio ao desenvolvimento de aplicações educacionais móveis para o ensino e aprendizagem de fundamentos de programação**. Tese (Doutorado) — Universidade de São Paulo, 2019. Citado nas páginas 24, 26, 43, 55, 56, 68, 71, 72, 78, 79 e 111.

- MARK, E. A. **Service Oriented Architecture: A Planning and Implementation Guide for Business and Technology**. [S.l.]: Wiley Online Library, 2006. Citado nas páginas 35 e 37.
- MEDEIROS, F.; TEDESCO, P.; GOMES, A. Arquiteturas de suporte à aprendizagem colaborativa sensível ao contexto. In: **Anais Eletrônicos do III Simpósio Hipertexto e Tecnologias na Educação. Universidade Federal de Pernambuco**. [S.l.: s.n.], 2011. p. 1–19. Citado nas páginas 41 e 43.
- MIT, M. I. o. T. 2012. Disponível em: <<http://appinventor.mit.edu/>>. Acesso em: 20/03/2019. Citado na página 24.
- MOEBERT, T.; HÖFLER, J.; LUCKE, U. An authoring system for adaptive mobile learning content. In: IEEE. **2016 IEEE 16th International Conference on Advanced Learning Technologies (ICALT)**. [S.l.], 2016. p. 82–84. Citado nas páginas 26, 47, 48, 49, 62, 64 e 67.
- MONTEIRO, B. d. S.; OLIVEIRA, E. J. de; GOMES, A. S.; NETO, F. M. Youubi: Ambiente de aprendizagem ubíqua. In: **Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)**. [S.l.: s.n.], 2015. v. 26, n. 1, p. 111. Citado nas páginas 24, 26, 68 e 78.
- MORALES, R.; IGLER, B.; BÖHM, S.; CHITCHAPOKA, P. Context-aware mobile language learning. **Procedia Computer Science**, Elsevier, v. 56, p. 82–87, 2015. Citado na página 71.
- MULLER, G. Reference architecture primer. In: **proceedings 17th International World WideWeb Conference (WWW 2008)**. [S.l.: s.n.], 2008. Citado nas páginas 32 e 33.
- NAKAGAWA, E. Y. **Uma contribuição ao projeto arquitetural de ambientes de engenharia de software**. Tese (Doutorado) — Universidade de São Paulo, 2006. Citado nas páginas 25, 30, 33, 34, 65, 87 e 105.
- NAKAGAWA, E. Y.; FLÁVIO, O.; BECKER, M. Ramodel: A reference model for reference architectures. In: **Joint Working IEEE/IFIP Conference on Software Architecture and European Conference on Software Architecture (WICSA/ECSA)**. Helsinki, Finland: [s.n.], 2012. p. 297–301. Citado nas páginas 25, 102, 106, 111 e 156.
- NAKAGAWA, E. Y.; SASAKI, M. M.; MALDONADO, J. C. An aspect-oriented framework for software documentation: An example on testing. In: **CibSE**. [S.l.: s.n.], 2009. p. 225–238. Citado nas páginas 33 e 66.
- OGATA, H.; YANO, Y. Context-aware support for computer-supported ubiquitous learning. In: **Proceedings. the 2nd IEEE International Workshop on Wireless and Mobile Technologies in Education, 2004**. [S.l.: s.n.], 2004. p. 27–34. Citado na página 43.
- OLIVEIRA, L. B. R.; NAKAGAWA, E. Y. A service-oriented reference architecture for software testing tools. In: CRNKOVIC, I.; GRUHN, V.; BOOK, M. (Ed.). **Software Architecture**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011. p. 405–421. Citado nas páginas 89 e 94.
- OLIVEIRA, L. B. R. d. **Estabelecimento de uma arquitetura de referência orientada a serviços para ferramentas de teste de software**. Tese (Doutorado) — Universidade de São Paulo, 2010. Citado na página 77.
- OLSON, N.; NOLIN, J.; NELHANS, G. Semantic web, ubiquitous computing, or internet of things? – a macro-analysis of scholarly publications. **Journal of Documentation**, v. 71, 09 2015. Citado na página 40.

OSSHIO, M. Estabelecimento de uma arquitetura de referência orientada a serviços para repositórios de linhas de processos de negócio. **UNIVERSIDADE FEDERAL DO MATO GROSSO DO SUL – UFMS**, 2014. Citado nas páginas 89 e 94.

OSSHIO, M. **Estabelecimento de uma arquitetura de referência orientada a serviços para repositórios de linhas de processos de negócio**. Dissertação (Mestrado) — Universidade Federal do Mato Grosso do Sul, 2014. Citado nas páginas 77, 87 e 100.

PAPAZOGLU, M. P.; TRAVERSO, P.; DUSTDAR, S.; LEYMAN, F. Service-oriented computing: a research roadmap. **Int. J. Cooperative Inf. Syst.**, v. 17, n. 2, p. 223–255, 2008. Disponível em: <<https://doi.org/10.1142/S0218843008001816>>. Citado nas páginas 35, 36, 37, 38 e 39.

PASSOS, M. C. D. A. **Adaptação ao u-learning: incremento de técnicas de ensino-aprendizagem, desenvolvimento da sabedoria digital e alcance da aprendizagem significativa**. Dissertação (Mestrado) — Universidade Federal de Pernambuco, 2016. Citado nas páginas 70, 71 e 79.

PENG, H. e. a. Ubiquitous knowledge construction: Mobile learning re-defined and a conceptual framework. In: **Innovations in Education and Teaching International**. [S.l.: s.n.], 2009. p. 171–1836. Citado na página 42.

PERNAS, A. M. **Sensibilidade à Situação em Sistemas Educacionais na Web**. Tese (Doutorado) — Universidade Federal do Rio Grande do Sul (UFRGS), 2012. Citado nas páginas 44 e 128.

PINHEIRO, R. C. Desenvolvimento e avaliação de aplicativo para aprendizagem ubíqua utilizando a youubi api. Universidade Federal Rural do Semi-Árido, 2019. Citado nas páginas 24, 25, 26, 45, 66, 68, 71, 78, 79 e 80.

POPOLIN, Á.; FOFONCA, E. Da ubiquidade tecnológica à aprendizagem ubíqua: contribuições teórico-práticas e construções de sentido para a educação musical. **Razón y Palabra**, v. 21, n. 4_99, p. 347–362, 2017. Citado nas páginas 71 e 79.

POSLAD, S. **Ubiquitous computing: smart devices, environments and interactions**. [S.l.]: John Wiley & Sons, 2009. Citado nas páginas 37 e 38.

SHAW, M.; GARLAN, D. **Software Architecture: perspective on an emerging discipline**. [S.l.]: Prentice Hall, 1996. Citado nas páginas 25, 30 e 31.

SILVA, W. A. d. **Uma arquitetura para orquestração da distribuição de água no semiárido brasileiro baseada em internet das coisas e computação em nuvem**. Dissertação (Mestrado) — Universidade Federal de Pernambuco, 2017. Citado nas páginas 71, 79 e 129.

SOAD, G. W.; FIORAVANTI, M. L.; FALVO, V.; MARCOLINO, A.; FILHO, N. F. D.; BARBOSA, E. F. Reqml-catalog: The road to a requirements catalog for mobile learning applications. In: **IEEE. 2017 IEEE Frontiers in Education Conference (FIE)**. [S.l.], 2017. p. 1–9. Citado na página 66.

SPÍNOLA, R. O.; TRAVASSOS, G. H. Towards a framework to characterize ubiquitous software projects. **Information and Software Technology**, Elsevier, v. 54, n. 7, p. 759–785, 2012. Citado nas páginas 41 e 42.

TABUENCA, B.; KALZ, M.; TERNIER, S.; SPECHT, M. Mobile authoring of open educational resources for authentic learning scenarios. **Universal Access in the Information Society**, Springer, v. 15, n. 3, p. 329–343, 2014. Citado nas páginas 24 e 26.

TSIAKMAKI, M.; HARTONAS, C. Implementing the crop reference architecture: The crop learning object editor. In: CITESEER. **BCI (Local)**. [S.l.], 2013. p. 72. Citado nas páginas 57, 58, 70 e 71.

WEISER, M. The computer for the 21st century. **ACM SIGMOBILE mobile computing and communications review**, ACM New York, NY, USA, v. 3, n. 3, p. 3–11, 1999. Citado na página 40.

YU, Z.; ZHOU, X.; SHU, L. Towards a semantic infrastructure for context-aware e-learning. **Multimedia Tools Appl**, v. 47, p. 71–86, 2010. Citado nas páginas 24 e 43.

ZHAO, R.; WANG, J. Visualizing the research on pervasive and ubiquitous computing. **Scientometrics**, Akadémiai Kiadó, co-published with Springer Science+ Business Media BV . . . , v. 86, n. 3, p. 593–612, 2011. Citado nas páginas 26, 40, 58, 59, 63, 70 e 71.

ZHAO, X.; JIN, Q. A service-oriented architecture for context-aware ubiquitous learning delivery. In: **Proceedings of the 8th International Conference on Fuzzy Systems and Knowledge Discovery**. [S.l.: s.n.], 2011. Citado na página 71.

AVALIAÇÃO - ESPECIALISTA ARQUITETO

Avaliação com especialista em arquitetura de *software*

Objetivo: verificar se a documentação da arquitetura MOA-RA, representada por suas visões, descrições, requisitos e informações gerais, estão claras, relacionadas com a proposta da arquitetura e podem ajudar no desenvolvimento de ambientes *u-learning*.

Avaliadores: especialistas na área de arquitetura de *software*, com conhecimentos em arquitetura de referência, arquitetura orientada a serviços (SOA) ou computação ubíqua, e algum envolvimento em projetos, desenvolvimento, avaliação ou pesquisa relacionadas à arquitetura de *software*.

Questionário (*checklist*): composto por 23 questões com 5 opções de resposta cada. As questões estão divididas em dois *checklists*: (1) Aspectos gerais: trata das questões gerais da arquitetura; e (2) Aspectos arquiteturais: trata das questões relacionadas à arquitetura de referência e SOA, abordando questões de modelagem, requisitos, visões e descrições arquiteturais. Os *checklists* serão respondidos com base na documentação da arquitetura MOA-RA

Documentação arquitetural: (i) informações gerais; (ii) lista de requisitos arquiteturais; (iii) visões arquiteturais da MOA-RA (visão geral, visão de módulos, visão em tempo de execução e visão de implantação) e as respectivas descrições de cada visão.

Opções de resposta: cada questão apresenta uma afirmação sobre determinada característica da arquitetura e cinco opções de respostas objetivas e uma subjetiva. O avaliador deve marcar a resposta que representa o nível de concordância com a questão e escrever um(a) comentário/sugestão, se desejar (opcional). Cada opção de resposta equivale a uma pontuação, baseado na escala *Likert* (LIKERT, 1932), que são as seguintes: (1) Discordo Totalmente; (2) Discordo; (3) Não Concordo, Nem Discordo (indiferente); (4) Concordo; e (5) Concordo Totalmente.

Tempo estimado: 35 minutos

DOCUMENTAÇÃO DA ARQUITETURA MOA-RA

1. INFORMAÇÕES GERAIS DA ARQUITETURA

Nome: Arquitetura de referência orientada a serviços para ambientes de aprendizagem sensível ao contexto (MOA-RA)

Domínio: Educacional / Ambientes de aprendizagem ubíqua

Motivações e Justificativa: O desenvolvimento de ambientes de aprendizagem ubíqua (*u-learning*) é uma tarefa difícil e que consome tempo e esforços dos programadores, devido à complexidade em desenvolver aplicações que devem gerenciar muitas informações do contexto de diferentes alunos, identificar as constantes mudanças de contexto que ocorrem durante o processo de aprendizagem e prover serviços e conteúdos adaptados a diferentes tipos de contexto. O problema agrava-se para o desenvolvimento de ambientes educacionais de autoria de atividades sensíveis ao contexto, para dispositivos móveis, que devem implementar mecanismos para a geração automática de código-fonte, na plataforma *mobile*, ou seja, o ambiente gera um novo aplicativo móvel, sensível ao contexto, para cada atividade construída pelo autor (professor). Isso aumenta a complexidade do desenvolvimento de ambientes ubíquos e gera um novo problema, que é a necessidade do aluno instalar no seu dispositivo móvel todas as atividades criadas por seus professores. Diante deste cenário, as características das arquiteturas de referência e SOA podem ajudar a contornar parte dos problemas apresentados, fornecendo diretrizes para conduzir a construção de ambientes com maior flexibilidade para o reúso e interoperabilidade com outras aplicações.

Objetivo da arquitetura: Auxiliar o desenvolvimento de ambientes *u-learning* que possam utilizar funcionalidades fornecidas na forma de serviços web para prover a autoria de materiais de aprendizagem móvel e sensíveis ao contexto, além de outras funções. O objetivo é baseado nas características da arquitetura de referência e SOA, possibilitando ainda contribuir para o reúso e evolução dos ambientes instanciados.

2. REQUISITOS ARQUITETURAIS

Os requisitos da MOA estão divididos em três grupos de requisitos arquiteturais relacionados aos seguintes domínios de aplicação: (GEE) requisitos dos ambientes educacionais genéricos; (ULE) requisitos da aprendizagem ubíqua; e (SOA) requisitos da arquitetura SOA.

2.1. Requisitos dos ambientes educacionais genéricos

Tabela 1: Requisitos arquiteturais dos ambientes educacionais genéricos

Req.	Requisitos dos ambientes educacionais genéricos
GEE-1	A arquitetura deve apoiar o desenvolvimento de ambientes educacionais que forneçam mecanismos para gerenciar as informações dos cursos , permitindo executar operações de cadastro, consulta, alteração e exclusão dos dados.
GEE-2	A arquitetura deve apoiar o desenvolvimento de ambientes educacionais que possam gerenciar os usuários , fornecendo diferentes visões para usuários com perfis diferentes.
GEE-3	A arquitetura deve permitir o desenvolvimento de ambientes educacionais que ofereçam mecanismos de comunicação (síncrona e assíncrona) entre os usuários.
GEE-4	A arquitetura deve permitir o desenvolvimento de ambientes educacionais que tenham suporte para a autoria de conteúdos/materiais didáticos e avaliações .
GEE-5	A arquitetura deve permitir o desenvolvimento de ambientes educacionais que possam disponibilizar conteúdos/materiais didáticos .
GEE-6	A arquitetura deve permitir o desenvolvimento de ambientes educacionais que possibilitem o gerenciamento de conteúdos/materiais didáticos .
GEE-7	A arquitetura deve permitir o desenvolvimento de ambientes educacionais que possam fornecer as informações do ambiente educacional, documentação do curso e do sistema .
GEE-8	A arquitetura deve permitir o desenvolvimento de ambientes educacionais que possam automatizar o processo de avaliação do aluno, permitindo a execução e correção automatizada das provas.
GEE-9	A arquitetura deve permitir o desenvolvimento de ambientes educacionais que possam adequar conteúdos de aprendizagem a padrões educacionais .
GEE-10	A arquitetura deve permitir o desenvolvimento de ambientes educacionais que forneçam feedback da avaliação do aluno.
GEE-11	A arquitetura deve permitir o desenvolvimento de ambientes educacionais que ofereçam mecanismos de segurança , garantindo o controle e integridade das informações e recursos do ambiente educacional.

2.2. Requisitos da aprendizagem ubíqua

Tabela 2: Requisitos arquiteturais da aprendizagem ubíqua (*u-learning*)

Req.	Requisitos dos ambientes u-learning
ULE-1	A arquitetura deve permitir o desenvolvimento de ambientes educacionais que possam obter as informações do contexto do usuário (ex. localização, preferências do aluno, etc.).
ULE-2	A arquitetura deve permitir o desenvolvimento de ambientes educacionais que possam reagir às mudanças de contexto , executando alguma ação proativamente.
ULE-3	A arquitetura deve apoiar o desenvolvimento de ambientes educacionais que permitam a descoberta automática de recursos (serviços ou dispositivos) próximos da localização do usuário (ex. catálogo virtual de livros de uma biblioteca / dispositivos dos usuários próximos...).
ULE-4	A arquitetura deve permitir o desenvolvimento de ambientes educacionais que possam adaptar conteúdos/atividades ao contexto do aluno .
ULE-5	A arquitetura deve permitir o desenvolvimento de ambientes educacionais que ofereçam suporte ao conhecimento just-in-time , permitindo que a aprendizagem possa acontecer em qualquer lugar e a qualquer hora.
ULE-6	A arquitetura deve permitir o desenvolvimento de ambientes educacionais que ofereçam serviços personalizados ao contexto individual do aluno (recomendação e entrega de conteúdo).
ULE-7	A arquitetura deve permitir o desenvolvimento de ambientes educacionais que possam usar as informações do contexto do aluno para prover serviços e informações.
ULE-8	A arquitetura deve permitir o desenvolvimento de ambientes educacionais que tenham suporte para a autoria de atividades de aprendizagem sensíveis ao contexto .
ULE-9	A arquitetura deve permitir o desenvolvimento de ambientes educacionais que possam entregar conteúdos para o aluno (delivery service) , de acordo com sua localização (ex. exibir informações no dispositivo móvel quando o aluno estiver em determinado local).
ULE-10	A arquitetura deve permitir o desenvolvimento de ambientes educacionais que possam notificar o aluno quando novos conteúdos e atividades forem criadas.
ULE-11	A arquitetura deve permitir o desenvolvimento de ambientes educacionais que possam apoiar a aprendizagem com mobilidade , por meio do uso de dispositivos móveis.

2.3. Requisitos da arquitetura orientada a serviços (SOA)

Tabela 3: Requisitos arquiteturais da arquitetura SOA

Req.	Requisitos SOA
SOA-1	A arquitetura deve viabilizar o desenvolvimento de ambientes educacionais que forneçam descrições semânticas dos serviços, facilitando a descoberta e utilização uniforme dos serviços educacionais.
SOA-2	A arquitetura deve viabilizar o desenvolvimento de ambientes educacionais fornecendo mecanismos para a publicação e descoberta de serviços por outras aplicações.
SOA-3	A arquitetura deve prover mecanismos para que os ambientes educacionais possam orquestrar ou coreografar serviços compostos por processos de negócio .
SOA-4	A arquitetura deve apoiar o desenvolvimento de ambientes educacionais que possam se comunicar (ou interagir) com diferentes aplicações (ou dispositivos) de plataformas heterogêneas de hardware e software.
SOA-5	A arquitetura deve apoiar o desenvolvimento de ambientes educacionais que possam incorporar novos serviços e ferramentas educacionais , permitindo a evolução do ambiente.
SOA-6	A arquitetura deve fornecer mecanismos para que o ambiente educacional possa disponibilizar as descrições do ambiente, políticas de uso, normas e qualquer documentação referente ao ambiente educacional.
SOA-7	A arquitetura deve permitir o desenvolvimento de ambientes educacionais, a partir da sua instanciação total ou parcial .
SOA-8	A arquitetura deve fornecer mecanismos para monitorar o atendimento aos requisitos e padrões definidos para os serviços cliente e servidor.

3. VISÕES ARQUITETURAIS

A arquitetura foi construída com base nos modelos Cliente-servidor e arquitetura em camadas, adotando o estilo arquitetural SOA e o padrão MVC (*Model-View-Controller*). Já as visões arquiteturais da MOA-RA estão representadas em UML (*Unified Modeling Language*).

Foram construídas quatro visões arquiteturais para documentar e representar a MOA-RA graficamente, seguindo o modelo ProSA-RA [NAKAGAWA, 2006] adotado para o desenvolvimento da arquitetura de referência proposta: (1) Visão Geral; (2) Visão de Módulo; (3) Visão em Tempo de Execução; e (4) Visão de Implantação.

Visões arquiteturais são artefatos importantes para facilitar a compreensão sobre a arquitetura, representando de forma gráfica aspectos da estrutura e organização dos componentes e seus relacionamentos, entre outros.

A seguir, são apresentadas as visões arquiteturais da MOA-RA acompanhadas pela descrição textual de cada uma delas.

3.1 Visão Geral

A visão geral é uma abstração visual de alto nível que demonstra os principais aspectos de uma arquitetura, em termos de estrutura, componentes, relacionamentos e a forma como eles estão organizados, sem se aprofundar nos detalhes mais específicos da arquitetura.

A Figura 1 mostra a visão geral da MOA-RA, seguindo o modelo cliente/servidor e composta por cinco camadas com seus respectivos módulos. Cada módulo trata de questões específicas e comuns a ele, e pode se relacionar com os demais módulos internos da camada e/ou com os módulos das outras camadas da arquitetura. A figura ainda mostra os relacionamentos entre os módulos da camada de Aplicação representados por setas sólidas, indicando relação entre módulos comuns, e por setas tracejadas para indicar relacionamento com um módulo transversal.

De acordo a Figura 1, pode-se observar que a MOA-RA propõe três tipos de aplicações distintas, do lado cliente, e possui cinco camadas do lado servidor: (1) Camada de Intermediação; (2) Camada de Qualidade de Serviços; (3) Camada de Apresentação; (4) Camada de Aplicação; e (5) Camada de Persistência (Dados).

A visão geral da MOA-RA e as descrições detalhadas das camadas e módulos da arquitetura são apresentadas a seguir.

Figura 1: Visão Geral da arquitetura MOA-RA

A Figura 2 apresenta uma visão ampliada da camada de Aplicação, camada onde estão concentrados os principais serviços educacionais da MOA-RA, para demonstrar o conjunto de serviços tratados por cada módulo da camada de Aplicação.

Figura 2: Visão estendida dos módulos da Camada de Aplicação da MOA-RA

Lado cliente da arquitetura

O lado cliente representa a aplicação do usuário (front-end) e é composto por três aplicações distintas: (i) Plataforma Web; (ii) Plataforma de Autoria; e (iii) Plataforma de Aprendizagem. A plataforma web pode ser executada em computadores (*desktop*) ou qualquer dispositivo com navegador e conexão à internet. Já as plataformas de autoria e aprendizagem são aplicativos de dispositivos móveis.

(i) Plataforma Web: É a principal aplicação do lado cliente, sendo responsável pelo gerenciamento de todas as informações, serviços e materiais didáticos produzidos no ambiente educacional, bem como dos recursos criados nas outras aplicações cliente (plataforma de autoria e plataforma de aprendizagem).

(ii) Plataforma de autoria (aplicação do professor): É o ambiente utilizado para criar atividades de aprendizagem sensível ao contexto, que serão posteriormente executadas na aplicação do aluno (plataforma de aprendizagem).

(iii) Plataforma de aprendizagem (aplicação do aluno): É o ambiente de execução das atividades criadas na plataforma de autoria, servindo como uma base única para execução de todas as atividades criadas pelo professor

Lado Servidor (Camadas da arquitetura)

O lado servidor é formado por 4 camadas (Intermediação, Apresentação, Aplicação e Persistência) que tratam de aspectos específicos e são responsáveis por um conjunto de funcionalidades para ambientes educacionais genéricos e ubíquos, fornecidas na forma de serviços web.

Camada de Intermediação (*intermediation layer*): Responsável por controlar as requisições enviadas entre serviços clientes e provedor, e fornecer mecanismos para a descoberta, publicação e disponibilização dos serviços educacionais do ambiente, entre outros. A camada é composta pelos seguintes elementos:

- Registro de Serviços: responsável pela publicação das informações dos serviços do ambiente, permitindo que possam ser descobertos e utilizados por outros serviços ou aplicações. Pode ser concretizado por meio de *brokers* ou *matchmakers*;

- **Agente de Serviço:** é o elemento mediador da comunicação entre cliente e servidor, atuando como arcabouço de roteamento e transporte das requisições aos serviços disponibilizados no ambiente educacional. O Agente de Serviço pode ser efetivado por meio de um ESB (*Enterprise Service Bus*);
- **Escalonador de Serviço:** verifica e processa as pendências das requisições aos serviços feitas pela aplicação cliente, descritas por uma linguagem padrão (ex. XML). No caso das requisições a serviços compostos, o Escalonador de Serviços analise também suas políticas de uso e requisitos de interoperabilidade necessários para a composição dos serviços solicitados.

Camada de Qualidade de Serviço (*quality of service layer*): responsável por verificar se as demais camadas provedoras de serviços estão cumprindo os requisitos de qualidade estabelecidos, atuando diretamente sob a camada de intermediação e parte da camada de apresentação (descrição dos serviços).

Camada de Apresentação (*presentation layer*): Responsável pela apresentação visual das informações, fornecendo a interface gráfica no estilo apropriado para o receptor da requisição. A camada é formada por três elementos, sendo eles:

- **Descrição do Serviço:** define os tipos de dados usados na requisição de serviços e resultado retornado ao cliente;
- **Engine de Serviço:** processa as requisições cliente convertendo a mensagem para o tipo de dado utilizado pelos serviços da Camada de Aplicação; e
- **Controlador:** responsável por orquestrar a execução dos serviços solicitados pela requisição.

Camada de Aplicação (*application layer*): É a principal camada da arquitetura onde estão concentrados os serviços educacionais que fornecem as principais funcionalidades para ambientes de aprendizagem sensíveis ao contexto. A camada possui 6 módulos responsáveis por tratar diferentes tipos de funcionalidade que também podem ser consumidas por outras aplicações educacionais. Os módulos presentes na Camada de Aplicação são descritos a seguir.

- **Módulo de Suporte:** incorpora 3 módulos que tratam de serviços básicos, não relacionados diretamente com questões de ensino e aprendizagem, mas apóiam o ambiente educacional na execução de algumas tarefas importantes. Foram definidos os seguintes módulos de suporte:
 - **Acesso:** responsável por controlar o acesso ao ambiente e aos serviços que ele oferece (ex. *login*, recuperação de senhas, chave de acesso);
 - **Documentação:** responsável por toda a documentação disponibilizada aos usuários do sistema, fornecendo mecanismos para o armazenamento e gerenciamento da documentação técnica (manual do uso, ajuda online, informações do sistema, etc.) e educacional (plano de aula, informações do curso, registro de atividades);
 - **Comunicação:** trata dos diferentes tipos de comunicação que podem ser utilizados pelo ambiente educacional, provendo mecanismos de suporte para comunicação síncrona e assíncrona, e ferramentas colaborativas (ex. SMS, mensagem de voz, *e-mail*, chamada de vídeo, videoconferência, fóruns, grupos, chat e *wikis*).
- **Módulo de Administração:** relacionado às questões administrativas do ambiente educacional, tais como cadastro do curso/usuário, matrícula de alunos, encargos docente, relatórios, ficha do aluno, informações do curso, entre outros. Basicamente, o módulo trata das seguintes questões: (1) Administração de curso: responsável pelo gerenciamento das informações relacionadas ao curso e disciplinas; e (2) Administração de usuários:

trata especificamente dos dados do usuário, como perfil, dados pessoais, permissões, disciplinas matriculadas, etc.

- **Módulo de Ensino:** trata das questões relacionadas à disponibilização, apresentação e entrega dos materiais didáticos (conteúdos e atividades u-learning) criados no ambiente, e também é responsável pela avaliação do aluno permitindo a aplicação de provas *online*, exercícios, etc.
- **Módulo de Autoria:** fornece apoio para a criação de materiais didáticos (conteúdos e avaliações) e atividades de aprendizagem. O módulo trata das seguintes questões: (1) Conteúdo: permite a criação e edição de conteúdos didáticos em diferentes mídias (texto, imagem, áudio, vídeo) e materiais de avaliação *online* (provas, exercícios, questionários, teste de múltipla escolha, etc); (2) Atividade context-aware: permite a criação de atividades de aprendizagem de campo, sensíveis ao contexto da localização. Essas atividades utilizam os conteúdos criados anteriormente (ex. texto), integradas de alguma forma com as coordenadas de localização capturadas pelo Módulo de Gerência do Contexto para criar um arquivo (“Atividade”) que será disponibilizado para plataforma de aprendizagem (cliente/aluno); e (3) Captura de conteúdo: trata da obtenção e armazenamento das interações do aluno na execução das atividades. Serve como um mecanismo que pode ser inserido na atividade para que ela possa salvar o feedback do aluno.
- **Módulo de Gerência do Contexto:** trata das questões relacionadas com aquisição de informações do contexto, mais especificamente ao contexto da localização geográfica (coordenadas de latitude e longitude – GPS), descoberta automática do contexto e serviços baseados na localização (LBS). O módulo é importante para a proposta da arquitetura, pois fornece um elemento fundamental para a composição e definição do comportamento das atividades sensíveis ao contexto, tais como os serviços LBS (proximidade, distância, mapa, trajeto, etc.)
- **Módulo de Adaptação:** responsável por mecanismos que permitem a adaptação do ambiente às características individuais do usuário ou do contexto onde está ocorrendo a aprendizagem. Baseado no perfil do aluno ou preferências de estudo, o ambiente pode passar a recomendar materiais mais adequados às suas características, ou ainda, ajustar as configurações do dispositivo para adaptar-se ao contexto do ambiente externo (ex. exibir somente conteúdos de áudio ou vídeo se o usuário estiver em movimento, ajustar o brilho de acordo com a luminosidade do local, entre outros).

Vale destacar os relacionamentos entre os principais módulos da camada de aplicação (autoria, ensino e contexto), que ocorrem da seguinte forma: o módulo de autoria usa serviços do módulo de contexto para criar atividades *context-aware*, e com o módulo de ensino que disponibiliza as atividades aos alunos.

Camada de Persistência (*persistence layer*): responsável pelo armazenamento e recuperação dos dados produzidos no ambiente educacional e solicitados para uso, respectivamente. A camada está localizada no servidor de banco de dados e incorpora um módulo de persistência para manter a integridade dos dados. O Módulo de Persistência é responsável pelo gerenciamento do banco de dados, realizando um conjunto de tarefas básicas como armazenamento, consulta, alteração e exclusão dos dados. Para executar essas tarefas pode-se utilizar um Sistema de Gerenciamento de Banco de Dados (SGBD).

Por fim, o conjunto de serviços ortogonais são serviços transversais que podem se comunicar com todos os serviços dos demais módulos. Tais serviços estão relacionados à persistência, segurança, configuração, responsividade, entre outros. O serviço de Segurança tem um papel importante para qualquer tipo de ambiente fornecendo mecanismos que ajudam a controlar o acesso aos serviços e registrar as atividades de cada seção do usuário para, caso necessário, possibilitar auditoria no sistema (autenticação, *logs* e registro de atividades).

Em síntese, o modelo de autoria sensível ao contexto proposto pela arquitetura MOA-RA pode oferecer as seguintes vantagens:

- A ferramenta de autoria não gera uma aplicação para cada atividade criada, eliminando a necessidade de instalar as atividades novas (ou atualizadas) e garantindo a integridade das atividades.
- A implementação desse modelo de autoria é aparentemente mais simples que das ferramentas de autoria com geração automática de código, pois não exige conhecimentos adicionais em Engenharia Dirigida por Modelos (MDE/MDD). Isto também facilita o desenvolvimento de atividades sensíveis ao contexto para professores que, em geral, não são da área de TI (Tecnologia da Informação).

3.2. Visão de Módulos

Figura 3: Visão de módulos da MOA-RA

A visão de módulo é utilizada para representar a arquitetura de referência em relação à forma como os módulos estão organizados e relacionados entre si, por meio de pacotes, sub-pacotes, classes e interfaces, fornecendo uma visão ampla dos elementos que implementam as funcionalidades dentro de cada módulo que compõe a arquitetura.

A visão de módulo da MOA foi construída com base nos requisitos arquiteturais e nas camadas da visão geral. As camadas são representadas por um pacote de serviços correspondente, exceto a camada de Intermediação que está representada pelo pacote Mediator de Serviço e pelo serviço “Registro Serviço”, que estão separados por questões de *design* gráfico para tornar a representação visual da visão de módulos mais simples e fácil de ser compreendida. A Tabela 4 apresenta os relacionamentos dos pacotes, sub-pacotes e serviços com as respectivas camadas da MOA-RA.

Tabela 4 – pacotes e serviços das camadas da MOA

Camada	Pacote	Sub-pacote	Serviço
1	Cliente	---	---
2	Registro Serviço	---	Publicar/localizar serviço
	Mediator de Serviço	AgenteServiço	Mediar serviço cliente/provedor
		EscalonadorServiço	Processar requisição serviço (verifica a conformidade aos padrões/políticas especificados)
3	Apresentação	EngineDeServiço	Processar requisição serviço (converte os dados para o modelo utilizado pelos serviços)
		Controlador	Tratar requisição de serviço
4	Suporte	Acesso	Login, recuperação de senhas e chave de acesso
		Comunicação	Enviar notificação aos usuários (ex. e-mail), Ferramentas de comunicação (ex. mensagem de texto, chat, fórum, wiki...)
		Documentação	Gerenciar a documentação do sistema, curso e usuário (manuais, ajuda online, plano de aula, histórico, registro de atividades..)
	Adm.	Adm. de curso	Operações básicas (CRUD – create, read, update, delete) relativas aos cursos oferecidos

		Adm. de usuário	Operações básicas (CRUD) relativas aos usuários do sistema
	Ensino	Disponibilização/ Apresentação	Gerenciar conteúdos/atividades, notificar usuário (novo cont.)
		Entrega	Enviar/exibir conteúdo para o usuário (ex. e-mail, dispositivo,...)
		Avaliação	Prova online, quizz, exercícios interativos, escolha, etc.
	Autoria	Atividade Sensível ao contexto	Operações básicas (CRUD), definir comportamento (relação AOE)
		Conteúdo	Operações básicas (CRUD)/conteúdo (texto, áudio, imagem..)
		Captura de conteúdo	Coletar e armazenar feedback do aluno durante a avaliação e conteúdos exigidos nas atividades (anotações, fotos, etc)
	Gerência do Contexto	Aquisição	Obter dados da localização (GPS)
		Descoberta	Identificar mudança de contexto / recursos próximos
		LBS	Proximidade, distância, mapa, trajetória
	Adaptação	Adapt. ao usuário	Uso de Ontologia ou Web semântica
		Adapt. ao contexto	Uso de Ontologia ou Web semântica
5	Persistência	---	Garantir a persistência dos dados do sistema (SGBD)
	Segurança	---	Autenticação, <i>logs</i> e registro de atividades

Legenda:

1 = Aplicação Cliente

2 = Camada de Intermediária

3 = Camada de Apresentação

4 = Camada de Aplicação

5 = Serviços Ortogonais

3.3. Visão em Tempo de Execução

Figura 4: Visão em Tempo de Execução da MOA-RA

A visão em tempo de execução representa a estrutura do ambiente educacional em execução demonstrando os componentes, interfaces (fornecidas e consumidas), interações, protocolos, tecnologias adotadas, entre outros. Os componentes representam os pacotes da visão de módulos.

Na visão em tempo de execução da MOA (Fig. 4) o componente *Cliente* representa o serviço da aplicação cliente (Web/App) que consome os serviços educacionais do provedor web. Este componente pode descobrir o endereço dos serviços que pretende utilizar pesquisando as informações no *RegistroServiço*, através do protocolo padrão definido (ex. UDDI). Uma vez descoberta a localização do serviço e sua respectiva descrição, o serviço *Cliente* pode estabelecer comunicação direta com a *EngineServiço*, que envia a requisição para o *Controlador* encaminhar para o serviço solicitado pelo cliente, ou comunicação indireta, através do *AgenteServiço* da Camada de Intermediação.

O componente *QualidadeServiço* é responsável por monitorar o cumprimento dos requisitos de qualidade do ambiente educacional, durante a comunicação entre os serviços cliente e provedor.

Já o componente *Ortogonalis* incorpora os serviços que podem ser consumidos por qualquer serviço da camada de Aplicação durante a execução do ambiente, caracterizando-se como serviços ortogonais (ex. persistência (SGBD), segurança (logs, registro de atividades, autenticação, etc.)).

3.4. Visão de Implantação

Figura 5: Visão de Implantação da MOA-RA

A visão de implantação representa o layout do ambiente físico onde o sistema será instalado, identificando todos os recursos de hardware, conexões e a parte do software que deve executar em cada parte do hardware que compõe o sistema. Os recursos de hardware são os dispositivos móveis, computadores, *Datacenter*, etc.

A visão de implantação da MOA-RA incorpora os seguintes elementos:

- Interface Cliente Web: permite o acesso ao ambiente educacional, por meio de um navegador instalado na máquina do cliente;
- Interface Cliente Móvel: permite o acesso ao ambiente educacional, por meio de um aplicativo (app) instalado no dispositivo móvel do cliente;
- Servidor Web: responsável por responder às requisições cliente (navegador e app) e fornecer acesso aos serviços do ambiente educacional (ex: *Tomcat*, *Apache*...);
- Servidor de Aplicação: contém todos os serviços educacionais do ambiente (autoria, ensino, gerência do contexto, adaptação, entre outros) e processa os serviços requisitados pelos clientes, utilizando protocolos SOAP e WSDL (ex. *JBoss*);
- Servidor de Agente de Serviço: armazena as aplicações relacionadas ao funcionamento do agente de serviços (ex. *ESB*);
- Registro de Serviço: responsável pelo gerenciamento das informações dos serviços, permitindo que eles possam ser localizados pelos serviços clientes;
- Repositório de Registros: responsável pelo armazenamento das informações dos registros de cada serviço fornecido pelo ambiente educacional;
- Servidor do Banco de Dados: representa o banco de dados onde são armazenadas todas as informações geradas no ambiente, assim como os serviços educacionais providos para as aplicações clientes. O gerenciamento dos dados e serviços armazenados no banco pode ser feito utilizando um SGBD (ex. *MySQL* e *PostgreSQL*).

Vale ressaltar que, todos os servidores presentes na visão de implantação da MOA-RA, ilustrados separadamente como máquinas distintas (servidor Web, aplicação, agente de serviço e banco de dados), podem estar fisicamente na mesma máquina.

Checklist 1 - Versão Arquiteto de Software

Aspectos gerais

1) As motivações para o desenvolvimento de MOA-RA são claros e compreensíveis (consultar item 1).

☐ - Concordo Totalmente

☐ - Concordo

☐ - Não Concordo, Nem discordo (indiferente)

☐ - Discordo

☐ - Discordo Totalmente

Comentários/sugestões: _____

2) O objetivo da arquitetura foi definido baseado nos problemas apresentados (consultar item 1)

☐ - Concordo Totalmente

☐ - Concordo

☐ - Não Concordo, Nem discordo (indiferente)

☐ - Discordo

☐ - Discordo Totalmente

Comentários/sugestões: _____

3) Os requisitos da MOA-RA são claros e estão dentro da proposta (consultar item 2).

☐ - Concordo Totalmente

☐ - Concordo

☐ - Não Concordo, Nem discordo (indiferente)

☐ - Discordo

☐ - Discordo Totalmente

Comentários/sugestões: _____

4) A visão geral da MOA-RA e suas respectivas descrições ajudam a compreender as características e proposta geral da arquitetura (consultar item 3.1).

☐ - Concordo Totalmente

☐ - Concordo

☐ - Não Concordo, Nem discordo (indiferente)

☐ - Discordo

☐ - Discordo Totalmente

Comentários/sugestões: _____

5) Observando as descrições e a representação gráfica da arquitetura é possível constatar que a MOA-RA incorpora os requisitos arquiteturais especificados (consultar item 3.1).

☐ - Concordo Totalmente

☐ - Concordo

☐ - Não Concordo, Nem discordo (indiferente)

☐ - Discordo

☐ - Discordo Totalmente

Comentários/sugestões: _____

6) A arquitetura incorpora elementos que tratam da segurança dos dados e usuários do ambiente educacional (consultar item 3.1).

☐ - Concordo Totalmente

☐ - Concordo

☐ - Não Concordo, Nem discordo (indiferente)

☐ - Discordo

☐ - Discordo Totalmente

Comentários/sugestões: _____

7) A arquitetura incorpora elementos relacionados às formas de comunicação que o ambiente é capaz de oferecer para os usuários (consultar item 3.1).

☐ - Concordo Totalmente

☐ - Concordo

☐ - Não Concordo, Nem discordo (indiferente)

☐ - Discordo

☐ - Discordo Totalmente

Comentários/sugestões: _____

8) Há indícios na documentação da MOA-RA que demonstram a viabilidade da solução proposta pela arquitetura (consultar item 3.1).

☐ - Concordo Totalmente

☐ - Concordo

☐ - Não Concordo, Nem discordo (indiferente)

☐ - Discordo

☐ - Discordo Totalmente

Comentários/sugestões: _____

Checklist 2 - Versão Arquiteto de Software

Aspectos de Arquiteturais

1) Considerando o conjunto de informações, requisitos, descrições e visões arquiteturais apresentados pode-se afirmar que a documentação da MOA-RA é capaz de apoiar o desenvolvimento de ambientes de aprendizagem ubíqua

☐ - Concordo Totalmente

☐ - Concordo

☐ - Não Concordo, Nem discordo (indiferente)

☐ - Discordo

☐ - Discordo Totalmente

Comentários/sugestões: _____

2) As visões arquiteturais estão intimamente relacionadas entre si, ou seja, foram modeladas para representar diferentes aspectos da arquitetura respeitando a padronização da estrutura, estilo, componentes e relacionamentos definidos na MOA-RA (consultar item 3).

☐ - Concordo Totalmente

☐ - Concordo

☐ - Não Concordo, Nem discordo (indiferente)

- ☐ - Discordo
☐ – Discordo Totalmente

Comentários/sugestões: _____

3) A arquitetura representa os aspectos transversais e os descreve de forma clara e compreensível (consultar item 3).

- ☐ - Concordo Totalmente
☐ - Concordo
☐ - Não Concordo, Nem discordo (indiferente)
☐ - Discordo
☐ – Discordo Totalmente

Comentários/sugestões: _____

4) A arquitetura foi modelada formalmente adotando terminologias, regras de relacionamento, notações e diagramas já reconhecidos e utilizados pela comunidade da engenharia de software (consultar item 3).

- ☐ - Concordo Totalmente
☐ - Concordo
☐ - Não Concordo, Nem discordo (indiferente)
☐ - Discordo
☐ – Discordo Totalmente

Comentários/sugestões: _____

5) A modelagem da arquitetura MOA-RA fornece o apoio necessário para a instanciação de arquiteturas concretas de ambientes *u-learning* (consultar item 3).

- ☐ - Concordo Totalmente
☐ - Concordo
☐ - Não Concordo, Nem discordo (indiferente)
☐ - Discordo
☐ – Discordo Totalmente

Comentários/sugestões: _____

6) A arquitetura representa e descreve os aspectos da infraestrutura física do ambiente educacional, tais como dispositivos, servidores, comunicação, implantação, entre outros (consultar itens 3.3/3.4).

- ☐ - Concordo Totalmente
☐ - Concordo
☐ - Não Concordo, Nem discordo (indiferente)
☐ - Discordo
☐ – Discordo Totalmente

Comentários/sugestões: _____

7) A arquitetura apresenta e descreve os aspectos lógicos da aplicação como pacotes, módulos classes e interfaces (consultar itens 3.2/3.3).

- ☐ - Concordo Totalmente
☐ - Concordo
☐ - Não Concordo, Nem discordo (indiferente)
☐ - Discordo

☐ – Discordo Totalmente

Comentários/sugestões: _____

8) A arquitetura descreve sobre os aspectos necessários para estabelecer comunicação interna e externa ao ambiente educacional (consultar itens 3.3/3.4)

☐ - Concordo Totalmente

☐ - Concordo

☐ - Não Concordo, Nem discordo (indiferente)

☐ - Discordo

☐ – Discordo Totalmente

Comentários/sugestões: _____

9) A arquitetura representa e descreve os locais de persistência dos dados (item 3.4).

☐ - Concordo Totalmente

☐ - Concordo

☐ - Não Concordo, Nem discordo (indiferente)

☐ - Discordo

☐ – Discordo Totalmente

Comentários/sugestões: _____

10) A arquitetura de referencia incorpora elementos para monitorar o não cumprimento dos requisitos de qualidade estabelecidos entre serviços Cliente/Servidor (consultar item 3.1).

☐ - Concordo Totalmente

☐ - Concordo

☐ - Não Concordo, Nem discordo (indiferente)

☐ - Discordo

☐ – Discordo Totalmente

Comentários/sugestões: _____

11) A arquitetura de referencia apóia o desenvolvimento de ambientes de aprendizagem que forneçam orientações normativas de uso, por meio de descrições padronizadas (consultar itens 2.3/3.1).

☐ - Concordo Totalmente

☐ - Concordo

☐ - Não Concordo, Nem discordo (indiferente)

☐ - Discordo

☐ – Discordo Totalmente

Comentários/sugestões: _____

12) A arquitetura de referencia permite a instanciação parcial ou total de ambientes *u-learning* (consultar itens 2.3/3).

☐ - Concordo Totalmente

☐ - Concordo

☐ - Não Concordo, Nem discordo (indiferente)

☐ - Discordo

☐ – Discordo Totalmente

Comentários/sugestões: _____

13) A arquitetura de referência possibilita a interação direta ou indireta (via *ESB – Enterprise Service Bus*) entre os serviços dos módulos educacionais e serviços de outras aplicações de plataformas de hardware e software heterogêneas (consultar itens 2 e 3)

☐ - Concordo Totalmente

☐ - Concordo

☐ - Não Concordo, Nem discordo (indiferente)

☐ - Discordo

☐ – Discordo Totalmente

Comentários/sugestões: _____

14) A arquitetura de referencia permite que o ambiente educacional possa evoluir incorporando de novos serviços, de maneira incremental (consultar itens 2.3 e 3)

☐ - Concordo Totalmente

☐ - Concordo

☐ - Não Concordo, Nem discordo (indiferente)

☐ - Discordo

☐ – Discordo Totalmente

Comentários/sugestões: _____

15) A arquitetura de referencia permite que os serviços educacionais possam ser publicados, descobertos e reutilizados por outras aplicações, por meio do tratamento uniforme de tais serviços (consultar itens 2.3 e 3).

☐ - Concordo Totalmente

☐ - Concordo

☐ - Não Concordo, Nem discordo (indiferente)

☐ - Discordo

☐ – Discordo Totalmente

Comentários/sugestões: _____

AVALIAÇÃO - ESPECIALISTA DO DOMÍNIO EDUCACIONAL

Avaliação com especialista do domínio educacional

Objetivo: verificar a capacidade da arquitetura MOA-RA em fornecer apoio para o desenvolvimento de ambientes e serviços educacionais genéricos e *u-learning*, por meio de *checklist*.

Avaliadores: especialistas do domínio educacional com ênfase em planejamento, desenvolvimento e/ou utilização de ambientes educacionais genéricos e *u-learning*.

Questionário (*checklist*): composto por 19 questões com 5 opções de resposta cada. As questões estão divididas em dois *checklist*: (1) Aspectos gerais: trata das questões gerais da arquitetura e (2) Aspectos do domínio educacional: trata das questões relacionadas aos ambientes educacionais genéricos e ambientes de aprendizagem ubíqua. Os *checklists* serão respondidos com base na documentação da arquitetura MOA-RA

Documentação arquitetural: (i) Informações gerais; (ii) Lista de requisitos (arquitetura de ambientes educacionais genéricos e *u-learning*); e (iii) Visão geral da arquitetura MOA-RA (acompanhada por suas descrições arquiteturais).

Opções de resposta: cada questão apresenta uma afirmação sobre determinada característica da arquitetura e cinco opções de respostas objetivas e uma subjetiva. O avaliador deve marcar a resposta que representa o nível de concordância com a questão e escrever um(a) comentário/sugestão, se desejar (opcional). Cada opção de resposta equivale a uma pontuação, baseado na escala *Likert* (LIKERT, 1932), que são as seguintes: (1) Discordo Totalmente; (2) Discordo; (3) Não Concordo, Nem Discordo (indiferente); (4) Concordo; e (5) Concordo Totalmente.

Tempo estimado: 27 minutos

DOCUMENTAÇÃO DA ARQUITETURA MOA-RA

1. INFORMAÇÕES GERAIS DA ARQUITETURA

Nome: Arquitetura de referência orientada a serviços para ambientes de aprendizagem sensível ao contexto (MOA-RA)

Domínio: Educacional / Ambientes de aprendizagem ubíqua

Motivações e Justificativa: O desenvolvimento de ambientes de aprendizagem ubíqua (*u-learning*) é uma tarefa difícil e que consome tempo e esforços dos programadores, devido à complexidade em desenvolver aplicações que devem gerenciar muitas informações do contexto de diferentes alunos, identificar as constantes mudanças de contexto que ocorrem durante o processo de aprendizagem e prover serviços e conteúdos adaptados a diferentes tipos de contexto. O problema agrava-se para o desenvolvimento de ambientes educacionais de autoria de atividades sensíveis ao contexto, para dispositivos móveis, que devem implementar mecanismos para a geração automática de código-fonte, na plataforma *mobile*, ou seja, o ambiente gera um novo aplicativo móvel, sensível ao contexto, para cada atividade construída pelo autor (professor). Isso aumenta a complexidade do desenvolvimento de ambientes ubíquos e gera um novo problema, que é a necessidade do aluno instalar no seu dispositivo móvel todas as atividades criadas por seus professores. Diante deste cenário, as características das arquiteturas de referência e SOA podem ajudar a contornar parte dos problemas apresentados, fornecendo diretrizes para conduzir a construção de ambientes com maior flexibilidade para o reuso e interoperabilidade com outras aplicações.

Objetivo da arquitetura: Auxiliar o desenvolvimento de ambientes *u-learning* que possam utilizar funcionalidades fornecidas na forma de serviços web para prover a autoria de materiais de aprendizagem móvel e sensíveis ao contexto, além de outras funções. O objetivo é baseado nas características da arquitetura de referência e SOA, possibilitando ainda contribuir para o reuso e evolução dos ambientes instanciados.

2. REQUISITOS ARQUITETURAIS

Os requisitos da MOA estão divididos em dois grupos de requisitos arquiteturais de ambientes educacionais genéricos e *u-learning*: (GEE) requisitos dos ambientes educacionais genéricos e (ULE) requisitos da aprendizagem ubíqua.

2.1. Requisitos dos ambientes educacionais genéricos

Tabela 1: Requisitos arquiteturais dos ambientes educacionais genéricos

Req.	Requisitos dos ambientes educacionais genéricos
GEE-1	A arquitetura deve apoiar o desenvolvimento de ambientes educacionais que forneçam mecanismos para gerenciar as informações dos cursos , permitindo executar operações de cadastro, consulta, alteração e exclusão dos dados.
GEE-2	A arquitetura deve apoiar o desenvolvimento de ambientes educacionais que possam gerenciar os usuários , fornecendo diferentes visões para usuários com perfis diferentes.
GEE-3	A arquitetura deve permitir o desenvolvimento de ambientes educacionais que ofereçam mecanismos de comunicação (síncrona e assíncrona) entre os usuários.
GEE-4	A arquitetura deve permitir o desenvolvimento de ambientes educacionais que tenham suporte para a autoria de conteúdos/materiais didáticos e avaliações .
GEE-5	A arquitetura deve permitir o desenvolvimento de ambientes educacionais que possam disponibilizar conteúdos/materiais didáticos .
GEE-6	A arquitetura deve permitir o desenvolvimento de ambientes educacionais que possibilitem o gerenciamento de conteúdos/materiais didáticos .
GEE-7	A arquitetura deve permitir o desenvolvimento de ambientes educacionais que possam fornecer as informações do ambiente educacional, documentação do curso e do sistema .
GEE-8	A arquitetura deve permitir o desenvolvimento de ambientes educacionais que possam automatizar o processo de avaliação do aluno, permitindo a execução e correção automatizada das provas.
GEE-9	A arquitetura deve permitir o desenvolvimento de ambientes educacionais que possam adequar conteúdos de aprendizagem a padrões educacionais .
GEE-10	A arquitetura deve permitir o desenvolvimento de ambientes educacionais que forneçam feedback da avaliação do aluno.
GEE-11	A arquitetura deve permitir o desenvolvimento de ambientes educacionais que ofereçam mecanismos de segurança , garantindo o controle e integridade das informações e recursos do ambiente educacional.

2.2. Requisitos da aprendizagem ubíqua

Tabela 2: Requisitos arquiteturais da aprendizagem ubíqua (*u-learning*)

Req.	Requisitos dos ambientes u-learning
ULE-1	A arquitetura deve permitir o desenvolvimento de ambientes educacionais que possam obter as informações do contexto do usuário (ex. localização, preferências do aluno, etc.).
ULE-2	A arquitetura deve permitir o desenvolvimento de ambientes educacionais que possam reagir às mudanças de contexto , executando alguma ação proativamente.
ULE-3	A arquitetura deve apoiar o desenvolvimento de ambientes educacionais que permitam a descoberta automática de recursos (serviços ou dispositivos) próximos da localização do usuário (ex. catálogo virtual de livros de uma biblioteca / dispositivos dos usuários próximos...).
ULE-4	A arquitetura deve permitir o desenvolvimento de ambientes educacionais que possam adaptar conteúdos/atividades ao contexto do aluno .
ULE-5	A arquitetura deve permitir o desenvolvimento de ambientes educacionais que ofereçam suporte ao conhecimento just-in-time , permitindo que a aprendizagem possa acontecer em qualquer lugar e a qualquer hora.
ULE-6	A arquitetura deve permitir o desenvolvimento de ambientes educacionais que ofereçam serviços personalizados ao contexto individual do aluno (recomendação e entrega de conteúdo).
ULE-7	A arquitetura deve permitir o desenvolvimento de ambientes educacionais que possam usar as informações do contexto do aluno para prover serviços e informações.
ULE-8	A arquitetura deve permitir o desenvolvimento de ambientes educacionais que tenham suporte para a autoria de atividades de aprendizagem sensíveis ao contexto .
ULE-9	A arquitetura deve permitir o desenvolvimento de ambientes educacionais que possam entregar conteúdos para o aluno (delivery service) , de acordo com sua localização (ex. exibir informações no dispositivo móvel quando o aluno estiver em determinado local).
ULE-10	A arquitetura deve permitir o desenvolvimento de ambientes educacionais que possam notificar o aluno quando novos conteúdos e atividades forem criadas.
ULE-11	A arquitetura deve permitir o desenvolvimento de ambientes educacionais que possam apoiar a aprendizagem com mobilidade , por meio do uso de dispositivos móveis.

3. VISÃO GERAL DA MOA-RA

A visão geral é uma abstração visual de alto nível que demonstra os principais aspectos de uma arquitetura, em termos de estrutura, componentes, relacionamentos e a forma como eles estão organizados, sem se aprofundar nos detalhes mais específicos da arquitetura.

A Figura 1 mostra a visão geral da MOA-RA, seguindo o modelo cliente/servidor e composta por cinco camadas com seus respectivos módulos. Cada módulo trata de questões específicas e comuns a ele, e pode se relacionar com os demais módulos internos da camada e/ou com os módulos das outras camadas da arquitetura. A Figura 1 ainda mostra os relacionamentos entre os módulos da camada de Aplicação representados por setas sólidas, indicando relação entre módulos comuns, e por setas tracejadas para indicar relacionamento com um módulo transversal.

De acordo a Figura 1, pode-se observar que a MOA-RA propõe três tipos de aplicações distintas, do lado cliente, e possui cinco camadas do lado servidor: (1) Camada de Intermediação; (2) Camada de Qualidade de Serviços; (3) Camada de Apresentação; (4) Camada de Aplicação; e (5) Camada de Persistência (Dados).

A visão geral da MOA-RA e as descrições detalhadas das camadas e módulos da arquitetura são apresentadas a seguir.

Figura 1: Visão Geral da arquitetura MOA-RA

A Figura 2 apresenta uma visão ampliada da camada de Aplicação, camada onde estão concentrados os principais serviços educacionais da MOA-RA, para demonstrar o conjunto de serviços tratados por cada módulo da camada de Aplicação.

Figura 2: Visão estendida dos módulos da Camada de Aplicação da MOA-RA

Lado cliente da arquitetura (Fig. 1)

O lado cliente representa a aplicação do usuário (*front-end*) e é composto por três aplicações distintas: (i) Plataforma Web; (ii) Plataforma de Autoria; e (iii) Plataforma de Aprendizagem. A plataforma web pode ser executada em computadores (*desktop*) ou qualquer dispositivo com navegador e conexão à internet. Já as plataformas de autoria e aprendizagem são aplicativos de dispositivos móveis.

(i) Plataforma Web: É a principal aplicação do lado cliente, sendo responsável pelo gerenciamento de todas as informações, serviços e materiais didáticos produzidos no ambiente educacional, bem como dos recursos criados nas outras aplicações cliente (plataforma de autoria e plataforma de aprendizagem).

(ii) Plataforma de autoria (aplicação do professor): É o ambiente utilizado para criar atividades de aprendizagem sensível ao contexto, que serão posteriormente executadas na aplicação do aluno (plataforma de aprendizagem).

(iii) Plataforma de aprendizagem (aplicação do aluno): É o ambiente de execução das atividades criadas na plataforma de autoria, servindo como uma base única para execução de todas as atividades criadas pelo professor

Lado Servidor - Camadas da arquitetura (Fig. 1)

O lado servidor é formado por 4 camadas (Intermediação, Apresentação, Aplicação e Persistência) que tratam de aspectos específicos e são responsáveis por um conjunto de funcionalidades para ambientes educacionais genéricos e ubíquos, fornecidas na forma de serviços *web*.

No entanto, vale ressaltar que nesta versão da documentação da MOA-RA (visão geral) só serão apresentadas as descrições das camadas, módulos e serviços básicos para qualquer ambiente (qualidade, segurança e persistência) e relacionados diretamente ao domínio dos especialistas educacionais (camada de aplicação).

Descrição das camadas (MOA-RA)

Camada de Qualidade de Serviço (*quality of service layer*): responsável por verificar se as demais camadas provedoras de serviços estão cumprindo os requisitos de qualidade estabelecidos, atuando diretamente sob a camada de intermediação e parte da camada de apresentação (descrição dos serviços).

Camada de Aplicação (*application layer*): É a principal camada da arquitetura onde estão concentrados os serviços educacionais que fornecem as principais funcionalidades para ambientes de aprendizagem sensíveis ao contexto. A camada possui 6 módulos responsáveis por tratar diferentes tipos de funcionalidade que também podem ser consumidas por outras aplicações educacionais. Os módulos presentes na Camada de Aplicação são descritos a seguir.

- **Módulo de Suporte:** incorpora 3 módulos que tratam de serviços básicos, não relacionados diretamente com questões de ensino e aprendizagem, mas apóiam o ambiente educacional na execução de algumas tarefas importantes. Foram definidos os seguintes módulos de suporte:
 - **Acesso:** responsável por controlar o acesso ao ambiente e aos serviços que ele oferece (ex. *login*, recuperação de senhas, chave de acesso);
 - **Documentação:** responsável por toda a documentação disponibilizada aos usuários do sistema, fornecendo mecanismos para o armazenamento e gerenciamento da documentação técnica (manual do uso, ajuda online, informações do sistema, etc.) e educacional (plano de aula, informações do curso, registro de atividades);
 - **Comunicação:** trata dos diferentes tipos de comunicação que podem ser utilizados pelo ambiente educacional, provendo mecanismos de suporte para comunicação síncrona e assíncrona, e ferramentas colaborativas (ex. SMS, mensagem de voz, *e-mail*, chamada de vídeo, videoconferência, fóruns, grupos, *chat* e *wikis*).
- **Módulo de Administração:** relacionado às questões administrativas do ambiente educacional, tais como cadastro do curso/usuário, matrícula de alunos, encargos docente, relatórios, ficha do aluno, informações do curso, entre outros. Basicamente, o módulo trata das seguintes questões: (1) Administração de curso: responsável pelo gerenciamento das informações relacionadas ao curso e disciplinas; e (2) Administração de usuários: trata especificamente dos dados do usuário, como perfil, dados pessoais, permissões, disciplinas matriculadas, etc.
- **Módulo de Ensino:** trata das questões relacionadas à disponibilização, apresentação e entrega dos materiais didáticos (conteúdos e atividades *u-learning*) criados no ambiente, e também é responsável pela avaliação do aluno permitindo a aplicação de provas *online*, exercícios, etc.
- **Módulo de Autoria:** fornece apoio para a criação de materiais didáticos (conteúdos e avaliações) e atividades sensíveis ao contexto (*context-aware*). O módulo trata das seguintes questões: (1) Conteúdo: permite a criação e edição de conteúdos didáticos em diferentes mídias (texto, imagem, áudio, vídeo) e materiais de avaliação *online* (provas, exercícios, questionários, teste de múltipla escolha, etc); (2) Atividade *context-aware*: permite a criação de atividades de aprendizagem sensíveis ao contexto da localização, para aulas de campo. Essas atividades utilizam os conteúdos criados previamente (ex. texto), integradas de alguma forma com os dados de localização (coordenadas GPS), capturados e gerenciados pelo Módulo de Gerência do Contexto que será disponibilizado e utilizado na aplicação de plataforma de aprendizagem (aluno); e (3) Captura de conteúdo: trata da obtenção e armazenamento das interações do aluno na execução das atividades. Serve como um mecanismo que pode ser inserido na atividade para que ela possa salvar o *feedback* do aluno.
- **Módulo de Gerência do Contexto:** trata das questões relacionadas com aquisição de informações do contexto, mais especificamente ao contexto da localização geográfica (coordenadas de latitude e longitude – GPS), descoberta automática do contexto e serviços baseados na localização (LBS). O módulo é importante para a proposta da arquitetura, pois fornece um elemento fundamental para a composição e definição do

comportamento das atividades sensíveis ao contexto, tais como os serviços LBS (proximidade, distância, mapa, trajeto, etc.)

- **Módulo de Adaptação:** responsável por mecanismos que permitem a adaptação do ambiente às características individuais do usuário ou do contexto onde está ocorrendo a aprendizagem. Baseado no perfil do aluno ou preferências de estudo, o ambiente pode passar a recomendar materiais mais adequados às suas características, ou ainda, ajustar as configurações do dispositivo para adaptar-se ao contexto do ambiente externo (ex. exibir somente conteúdos de áudio ou vídeo se o usuário estiver em movimento, ajustar o brilho de acordo com a luminosidade do local, entre outros).

Vale destacar os relacionamentos entre os principais módulos da camada de aplicação (autoria, ensino e contexto), que ocorrem da seguinte forma: o módulo de autoria usa serviços do módulo de contexto para criar atividades *context-aware*, e com o módulo de ensino que disponibiliza as atividades aos alunos.

Camada de Persistência (*persistence layer*): responsável pelo armazenamento e recuperação dos dados produzidos no ambiente educacional e solicitados para uso, respectivamente. A camada está localizada no servidor de banco de dados e incorpora um módulo de persistência para manter a integridade dos dados. O Módulo de Persistência é responsável pelo gerenciamento do banco de dados, realizando um conjunto de tarefas básicas como armazenamento, consulta, alteração e exclusão dos dados. Para executar essas tarefas pode-se utilizar um Sistema de Gerenciamento de Banco de Dados (SGBD).

Por fim, o conjunto de serviços ortogonais são serviços transversais que podem se comunicar com todos os serviços dos demais módulos. Tais serviços estão relacionados à persistência, segurança, configuração, responsividade, entre outros. O serviço de Segurança tem um papel importante para qualquer tipo de ambiente fornecendo mecanismos que ajudam a controlar o acesso aos serviços e registrar as atividades de cada seção do usuário para, caso necessário, possibilitar auditoria no sistema (autenticação, *logs* e registro de atividades).

Em síntese, o modelo de autoria sensível ao contexto proposto pela arquitetura MOA-RA pode oferecer as seguintes vantagens:

- A ferramenta de autoria não gera uma aplicação para cada atividade criada, eliminando a necessidade de instalar as atividades novas (ou atualizadas) e garantindo a integridade das atividades.
- A implementação desse modelo de autoria é aparentemente mais simples que das ferramentas de autoria com geração automática de código, pois não exige conhecimentos adicionais em Engenharia Dirigida por Modelos (MDE/MDD). Isto também facilita o desenvolvimento de atividades sensíveis ao contexto para professores que, em geral, não são da área de TI (Tecnologia da Informação).

Checklist 1 - Versão Especialista Educacional

Aspectos gerais

1) As motivações para o desenvolvimento de MOA-RA são claros e compreensíveis (consultar item 1).

- ☐ - Concordo Totalmente
☐ - Concordo
☐ - Não Concordo, Nem discordo (indiferente)
☐ - Discordo
☐ - Discordo Totalmente

Comentários/sugestões: _____

2) O objetivo da arquitetura foi definido baseado nos problemas apresentados (consultar item 1)

- ☐ - Concordo Totalmente
☐ - Concordo
☐ - Não Concordo, Nem discordo (indiferente)
☐ - Discordo
☐ - Discordo Totalmente

Comentários/sugestões: _____

3) Os requisitos da MOA-RA são claros e estão dentro da proposta (consultar item 2).

- ☐ - Concordo Totalmente
☐ - Concordo
☐ - Não Concordo, Nem discordo (indiferente)
☐ - Discordo
☐ - Discordo Totalmente

Comentários/sugestões: _____

4) A visão geral da MOA-RA e suas respectivas descrições ajudam a compreender as características e proposta geral da arquitetura (consultar item 3).

- ☐ - Concordo Totalmente
☐ - Concordo
☐ - Não Concordo, Nem discordo (indiferente)
☐ - Discordo
☐ - Discordo Totalmente

Comentários/sugestões: _____

5) Observando as descrições e a representação gráfica da arquitetura é possível constatar que a MOA-RA incorpora os requisitos arquiteturais especificados (consultar item 3).

- ☐ - Concordo Totalmente
☐ - Concordo
☐ - Não Concordo, Nem discordo (indiferente)
☐ - Discordo
☐ - Discordo Totalmente

Comentários/sugestões: _____

6) A arquitetura incorpora elementos que tratam da segurança dos dados e usuários do ambiente educacional (consultar item 3).

☐ - Concordo Totalmente

☐ - Concordo

☐ - Não Concordo, Nem discordo (indiferente)

☐ - Discordo

☐ - Discordo Totalmente

Comentários/sugestões: _____

7) A arquitetura incorpora elementos relacionados às formas de comunicação que o ambiente é capaz de oferecer para os usuários (consultar item 3).

☐ - Concordo Totalmente

☐ - Concordo

☐ - Não Concordo, Nem discordo (indiferente)

☐ - Discordo

☐ - Discordo Totalmente

Comentários/sugestões: _____

8) Há indícios na documentação da MOA-RA que demonstram a viabilidade da solução proposta pela arquitetural (consultar item 3).

☐ - Concordo Totalmente

☐ - Concordo

☐ - Não Concordo, Nem discordo (indiferente)

☐ - Discordo

☐ - Discordo Totalmente

Comentários/sugestões: _____

Checklist 2 - Versão Especialista Educacional

Aspectos do Domínio Educacional

1) A arquitetura permite o desenvolvimento de ambientes educacionais que possam disponibilizar as informações dos diferentes cursos, turmas, disciplinas e usuários cadastrados no ambiente (consultar itens 2 e 3).

☐ - Concordo Totalmente

☐ - Concordo

☐ - Não Concordo, Nem discordo (indiferente)

☐ - Discordo

☐ - Discordo Totalmente

Comentários/sugestões: _____

2) A arquitetura permite o desenvolvimento de ambientes educacionais que controlem o acesso ao ambiente e materiais didáticos de acordo com o perfil do usuário (consultar itens 2 e 3).

☐ - Concordo Totalmente

☐ - Concordo

☐ - Não Concordo, Nem discordo (indiferente)

☐ - Discordo

☐ – Discordo Totalmente

Comentários/sugestões: _____

3) A arquitetura permite o desenvolvimento de ambientes educacionais que ofereçam mecanismos para autoria de avaliações e materiais didáticos (consultar itens 2 e 3).

☐ - Concordo Totalmente

☐ - Concordo

☐ - Não Concordo, Nem discordo (indiferente)

☐ - Discordo

☐ – Discordo Totalmente

Comentários/sugestões: _____

4) A arquitetura permite o desenvolvimento de ambientes educacionais com mecanismos para realizar avaliação do aluno, de forma automatizada ou *online* (consultar itens 2 e 3).

☐ - Concordo Totalmente

☐ - Concordo

☐ - Não Concordo, Nem discordo (indiferente)

☐ - Discordo

☐ – Discordo Totalmente

Comentários/sugestões: _____

5) A arquitetura permite o desenvolvimento de ambientes educacionais que forneçam *feedback* da avaliação para o aluno (consultar itens 2 e 3).

☐ - Concordo Totalmente

☐ - Concordo

☐ - Não Concordo, Nem discordo (indiferente)

☐ - Discordo

☐ – Discordo Totalmente

Comentários/sugestões: _____

6) A arquitetura permite o desenvolvimento de ambientes educacionais personalizados, ou seja, as configurações do ambiente (linguagem, configurações do computador ou dispositivo móvel, etc.) podem ser personalizadas pelo usuário ou pelo próprio ambiente de acordo com as características do usuário (consultar itens 2 e 3).

☐ - Concordo Totalmente

☐ - Concordo

☐ - Não Concordo, Nem discordo (indiferente)

☐ - Discordo

☐ – Discordo Totalmente

Comentários/sugestões: _____

7) A arquitetura permite o desenvolvimento de ambientes educacionais que possam identificar, obter e gerenciar as informações do contexto (consultar itens 2 e 3).

☐ - Concordo Totalmente

☐ - Concordo

☐ - Não Concordo, Nem discordo (indiferente)

☐ - Discordo

☐ – Discordo Totalmente

Comentários/sugestões: _____

8) A arquitetura permite o desenvolvimento de ambiente educacionais utilizar as informações do contexto para adaptar serviços e conteúdos de aprendizagem (consultar itens 2 e 3).

☐ - Concordo Totalmente

☐ - Concordo

☐ - Não Concordo, Nem discordo (indiferente)

☐ - Discordo

☐ – Discordo Totalmente

Comentários/sugestões: _____

9) A arquitetura permite o desenvolvimento de ambientes educacionais que ofereçam mecanismos para autoria de conteúdos de aprendizagem para aplicativos móveis sensíveis ao contexto da localização do usuário (consultar itens 2 e 3).

☐ - Concordo Totalmente

☐ - Concordo

☐ - Não Concordo, Nem discordo (indiferente)

☐ - Discordo

☐ – Discordo Totalmente

Comentários/sugestões: _____

10) A arquitetura permite o desenvolvimento de ambientes educacionais que possam disponibilizar acesso ao ambiente e materiais didáticos a qualquer hora e lugar que o usuário esteja (consultar itens 2 e 3).

☐ - Concordo Totalmente

☐ - Concordo

☐ - Não Concordo, Nem discordo (indiferente)

☐ - Discordo

☐ – Discordo Totalmente

Comentários/sugestões: _____

11) A arquitetura permite o desenvolvimento de ambientes educacionais que possam descobrir automaticamente novos serviços e usuários próximos (consultar itens 2 e 3).

☐ - Concordo Totalmente

☐ - Concordo

☐ - Não Concordo, Nem discordo (indiferente)

☐ - Discordo

☐ – Discordo Totalmente

Comentários/sugestões: _____

AVALIAÇÃO - PROTÓTIPO BATALHA NAVAL

QUESTIONÁRIO DE AVALIAÇÃO DO PROTÓTIPO (BATALHA NAVAL)

[illegible]

AVALIAÇÃO - PROTÓTIPO CLASS PATH

QUESTIONÁRIO DE AVALIAÇÃO DO PROTÓTIPO (CLASS PATH)

[illegible]

AVALIAÇÃO - PROTÓTIPO APP-AUTHOR

QUESTIONÁRIO DE AVALIAÇÃO DO PROTÓTIPO (APP-AUTHOR)

Item	Questões	Discordo Plenamente	Discordo	Neutro	Concordo	Concordo Plenamente	Score
01	Eu achei o layout do aplicativo simples e bem estruturado					4	4
02	Eu achei difícil encontrar todas as funcionalidades presentes no aplicativo.	4					4
03	Eu gostaria de usar esse aplicativo com maior frequência				1	3	3,7
04	Eu achei que o aplicativo utiliza uma linguagem complexa, com textos difíceis de serem compreendidos.	2	2				3,5
05	Eu achei fácil a navegação entre as telas do aplicativo.					4	4
06	Eu não me senti confiante em utilizar e recomendar o App-Author para outros professores.	2	2				3,5
07	Acho que o aplicativo pode contribuir para aprendizagem, pois ele permite a construção de materiais didáticos interativos, atraentes e capazes de melhorar a motivação do aluno para os estudos.					4	4
08	Eu não acho que o aplicativo possa motivar o professor, por facilitar o processo de construção de recursos didáticos ubíquos.	2	1		1		3
09	Acho que para criar atividades de aprendizagem sensíveis ao contexto, para dispositivos moveis, utilizando o aplicativo App-Author, o professor só precisa ter conhecimentos básicos em informática. ´					4	4
10	Eu não acho que as várias funções do aplicativo estavam bem integradas.	2	1	1			3,2
							92,2

AVALIAÇÃO - PROTÓTIPO APP-LEARN

QUESTIONÁRIO DE AVALIAÇÃO DO PROTÓTIPO (APP-LEARN)

Item	Questões	Discordo Plenamente	Discordo	Neutro	Concordo	Concordo Plenamente	Score
01	Eu achei bonito e organizado o layout das telas do aplicativo				1	3	3,7
02	Senti dificuldade para encontrar as funcionalidades do App-Learn	2	1		1		3
03	Gostaria de usar o aplicativo mais vezes			1		3	3,5
04	O App-Learn utiliza uma linguagem complexa na descrição das informações, com textos difíceis de serem compreendidos, a primeira vista	1	1	1		1	2
05	Achei fácil navegar entre as telas do App-Learn					4	4
06	Não me sinto confiante para indicar o App-Learn para outros alunos	2	1	1			3,2
07	Acho que o App-Learn ajuda a reforçar o conteúdo e melhorar a aprendizagem do assunto				2	2	3,5
08	Não me senti mais motivado(a) para estudar o assunto, utilizando o aplicativo	1	1	2			2,5
09	Acho que o App-Learn pode ser utilizado em outras disciplinas e áreas do conhecimento			1	1	2	3,2
10	As funcionalidades do aplicativo não estão funcionando corretamente	2	1	1			3,2
							79,5