

UNIVERSIDADE FEDERAL DE UBERLÂNDIA
FACULDADE DE COMPUTAÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO



**DISTRIBUIÇÃO E PERSONALIZAÇÃO DE CONTEÚDO
MULTIMÍDIA EM AMBIENTES EDUCACIONAIS UBÍQUOS**

RAFAEL DIAS ARAÚJO

Uberlândia - Minas Gerais

2013

UNIVERSIDADE FEDERAL DE UBERLÂNDIA
FACULDADE DE COMPUTAÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO



RAFAEL DIAS ARAÚJO

DISTRIBUIÇÃO E PERSONALIZAÇÃO DE CONTEÚDO MULTIMÍDIA EM AMBIENTES EDUCACIONAIS UBÍQUOS

Dissertação de Mestrado apresentada à Faculdade de Computação da Universidade Federal de Uberlândia, Minas Gerais, como parte dos requisitos exigidos para obtenção do título de Mestre em Ciência da Computação.

Área de concentração: Sistemas de Computação.

Orientador:

Prof. Dr. Renan Gonçalves Cattelan

Co-orientadora:

Prof^a. Dr^a. Sandra Aparecida de Amo

Uberlândia, Minas Gerais
2013

UNIVERSIDADE FEDERAL DE UBERLÂNDIA
FACULDADE DE COMPUTAÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

Os abaixo assinados, por meio deste, certificam que leram e recomendam para a Faculdade de Computação a aceitação da dissertação intitulada “**Distribuição e Personalização de Conteúdo Multimídia em Ambientes Educacionais Ubíquos**” por **Rafael Dias Araújo** como parte dos requisitos exigidos para a obtenção do título de **Mestre em Ciência da Computação**.

Uberlândia, 06 de Agosto de 2013

Orientador:

Prof. Dr. Renan Gonçalves Cattelan
Universidade Federal de Uberlândia

Co-orientadora:

Prof^a. Dr^a. Sandra Aparecida de Amo
Universidade Federal de Uberlândia

Banca Examinadora:

Prof^a. Dr^a. Maria da Graça Campos Pimentel
Universidade de São Paulo

Prof. Dr. Fabiano Azevedo Dorça
Universidade Federal de Uberlândia

UNIVERSIDADE FEDERAL DE UBERLÂNDIA
FACULDADE DE COMPUTAÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

Data: 06 de Agosto de 2013

Autor: **Rafael Dias Araújo**
Título: **Distribuição e Personalização de Conteúdo Multimídia em Ambientes Educacionais Ubíquos**
Faculdade: **Faculdade de Computação**
Grau: **Mestrado**

Fica garantido à Universidade Federal de Uberlândia o direito de circulação e impressão de cópias deste documento para propósitos exclusivamente acadêmicos, desde que o autor seja devidamente informado.

Autor

O AUTOR RESERVA PARA SI QUALQUER OUTRO DIREITO DE PUBLICAÇÃO DESTE DOCUMENTO, NÃO PODENDO O MESMO SER IMPRESSO OU REPRODUZIDO, SEJA NA TOTALIDADE OU EM PARTES, SEM A PERMISSÃO ESCRITA DO AUTOR.

Dedicatória

Aos meus pais Vilmar e Maria
Aos meus irmãos Eduardo e Daniel

Agradecimentos

À Deus, pelo dom da vida.

Ao meu orientador, Prof. Dr. Renan Gonçalves Cattelan, pelos ensinamentos, incentivo e amizade ao longo desse período.

À minha família, especialmente àqueles que, com carinho, sempre me apoiaram e incentivaram.

À Prof^a. Dr^a. Sandra Aparecida de Amo e ao Prof. Dr. Pedro Frosi Rosa, pelas sugestões e contribuições dadas no decorrer deste trabalho.

Ao secretário do PPGCC/UFU e também aos professores deste programa que contribuíram para a minha formação.

A todos os amigos, pelo companheirismo, incentivo e conselhos. Em especial, aos amigos do PPGCC/UFU pelo convívio diário e apoio tanto nos bons momentos quanto naqueles mais difíceis: Hiran, Taffarel, Evaldo, Igor, Rômerson, Elder, Ana Maria, Lufs, New, Geycy, Juliete, Cleiane, Fabíola, Joicy e Miguel.

À CAPES, pelo apoio financeiro.

Enfim, agradeço a todos que estiveram ao meu lado e me apoiaram, direta ou indiretamente, para a concretização deste trabalho.

*"The most profound technologies are those that disappear.
They weave themselves into the fabric of everyday life
until they are indistinguishable from it."
(Mark Weiser, 1991)*

Resumo

Computação ubíqua é conhecida como a terceira onda da computação, a qual visa tornar os dispositivos computacionais mais integrados à vida das pessoas. Ela tem ajudado as pessoas a realizarem suas atividades diárias em diferentes domínios, especialmente no domínio educacional, onde auxilia instrutores e alunos durante as suas atividades acadêmicas obtendo conhecimento em qualquer lugar e a qualquer momento, fazendo uso de formas não intrusivas para recuperar o conteúdo desejado. Captura e acesso multimídia e ciência de contexto são dois importantes temas em pesquisas de computação ubíqua. Enquanto o primeiro produz automaticamente um grande volume de mídias digitais (ex., vídeo, áudio, slides, comentários de texto), o segundo oferece meios adequados para integrar e acessar tal conteúdo. Ao fundir os dois conceitos, os usuários podem se beneficiar de ferramentas que lhes permitam recuperar, visualizar e interagir com os artefatos de mídia capturados. Assim, este trabalho apresenta um modelo de comunicação baseado em redes *peer-to-peer*, utilizado para transferência e replicação do conteúdo capturado. Além disso, uma arquitetura de acesso contextual é apresentada para personalização e recomendação de conteúdo multimídia interativo capturado em ambientes educacionais instrumentados. Essa arquitetura utiliza uma abordagem que considera informações de contexto, preferências do usuário e restrições de apresentação a fim de personalizar a experiência de acesso, adequando-a às necessidades dos usuários. Como estudo de caso, as propostas foram implementadas no *Classroom eXperience*, que é uma plataforma de computação ubíqua para captura multimídia concebida para gravar automaticamente as aulas ministradas e disponibilizá-las posteriormente aos alunos.

Palavras chave: captura e acesso, ciência de contexto, personalização de conteúdo multimídia, mídia digital interativa, ambientes educacionais instrumentados, redes *peer-to-peer*, computação ubíqua.

Abstract

Ubiquitous computing is known as the third wave of computing, which aims at making computational devices become more permeated to the users lives. It has helped people to perform their daily activities in many areas of expertise, especially in the educational domain where it assists instructors and students during their academic duties, achieving knowledge anywhere and anytime, making use of unobtrusive ways of reaching the desired content. Multimedia capture and context-awareness are two relevant subjects in ubiquitous computing research. While the former results in large amounts of digital media (e.g., video, audio, slides, text comments) being automatically produced, the latter offers the proper means to integrate and access such content. By merging the two concepts, users may benefit from tools that allow them to retrieve, view and interact with the captured media artifacts. Thus, this work presents a communication model based on peer-to-peer networks, used for transferring and replicating the captured content. Furthermore, a contextual access architecture is presented for personalization and recommendation of interactive multimedia content captured in an instrumented educational environment. This architecture takes into account context information, user preferences and presentation constraints in order to personalize the access experience, tailoring it to users' needs. As a case study, the proposals were implemented in *Classroom eXperience*, which is a ubiquitous computing platform for multimedia capturing designed to automatically record lectures and make them available to students.

Keywords: capture and access, context-awareness, multimedia content personalization, interactive digital media, instrumented classrooms, peer-to-peer networks, ubiquitous computing.

Sumário

Lista de Figuras	xix
Lista de Tabelas	xxi
Lista de Abreviaturas e Siglas	xxiii
1 Introdução	25
1.1 Contextualização	25
1.2 Motivação	26
1.3 Objetivos	28
1.4 Método de Pesquisa	28
1.5 Estrutura da Dissertação	29
2 Ambientes Educacionais Ubíquos: Fundamentos e Conceitos	31
2.1 Computação Ubíqua	32
2.1.1 Captura e Acesso	33
2.1.2 Características das Aplicações de Captura e Acesso	34
2.1.3 Ciência de Contexto	35
2.2 Ambientes Educacionais Ubíquos	37
2.3 Considerações Finais	37
3 Um Modelo para Distribuição de Conteúdo Multimídia Usando Redes	
<i>Peer-to-Peer</i>	39
3.1 Camada de Abstração para Transferência e Replicação de Conteúdo	40
3.2 Especificação e Projeto do Protocolo de Comunicação	41
3.2.1 Ambiente	42
3.2.2 Vocabulário	42
3.2.3 Formato	44
3.2.4 Serviços e Regras Procedimentais	45
3.3 Considerações Finais	51

4	Uma Arquitetura para Acesso Personalizado de Conteúdo Multimídia	53
4.1	Mapeamento das Dimensões de Contexto	54
4.2	Preferências Contextuais de Usuários	56
4.3	Restrições de Apresentação	57
4.4	Visão Geral da Arquitetura	58
4.5	Considerações Finais	60
5	Classroom eXperience: Estudo de Caso	61
5.1	Pré-produção	62
5.2	Gravação ao vivo	64
5.3	Pós-produção	64
5.4	Acesso	65
5.4.1	Aspectos de Usabilidade da Interface de Acesso e Apresentação . .	66
5.4.2	Personalização e Recomendação de Conteúdo Utilizando Contexto, Preferências e Restrições	71
5.5	Considerações Finais	78
6	Trabalhos Relacionados	79
7	Conclusão	87
7.1	Resultados e Contribuições	88
7.2	Limitações	88
7.3	Trabalhos Futuros	89
7.4	Considerações Finais	89
	Referências	91
A	Diagrama Entidade-Relacionamento (DER) da base de dados do CX	99
B	Restrições de Apresentação Utilizadas	101
A	Estrutura do arquivo <i>Learning Object Metadata</i> (LOM) armazenado	103

Lista de Figuras

2.1	Alguns exemplos de dispositivos de computação ubíqua.	32
2.2	Obtendo informações contextuais através de um dispositivo móvel (Business Computing World, 2011).	36
3.1	Localização da camada CAL.	40
3.2	Topologia de rede com <i>peers</i> produtores e consumidores.	42
3.3	Formato de primitiva do protocolo proposto.	45
3.4	Autômato do serviço <i>start_session</i>	46
3.5	Autômato do serviço <i>end_session</i>	46
3.6	Autômato do serviço <i>list_status</i>	47
3.7	Diagrama temporal do serviço <i>list_status</i>	48
3.8	Autômato para o serviço <i>ft_get</i>	49
3.9	Autômato para o serviço <i>ft_put</i>	50
4.1	Contexto de acesso do usuário.	55
4.2	Visão Geral da Arquitetura de Acesso Contextual (CAA).	59
5.1	Criação de uma nova aula no <i>Classroom eXperience</i>	63
5.2	Aula sendo capturada com a infraestrutura do <i>Classroom eXperience</i>	64
5.3	Protótipo da tela de listagem das aulas de uma disciplina.	67
5.4	Protótipo da tela de apresentação do conteúdo de uma aula capturada.	68
5.5	Página principal de listagem das disciplinas matriculadas no <i>Classroom eXperience</i>	70
5.6	<i>Popup</i> de exibição das turmas disponíveis para matrícula.	70
5.7	Documento HTML apresentando os <i>slides</i> e o vídeo de uma aula capturada.	71
5.8	Formulário Web utilizado para cadastrar uma nova preferência pessoal.	73
5.9	Página inicial de uma disciplina cadastrada no <i>Classroom eXperience</i>	74
5.10	<i>Pop-up</i> com informações gerais de uma aula e o <i>ranking</i> de apresentação (parte inferior).	77
5.11	Quantidade de acesso às aulas ministradas.	77

6.1	Arquitetura do sistema <i>Tele-Board</i> (Gumienny <i>et al.</i> , 2013).	81
6.2	Arquitetura do sistema proposto por Wang e Wu (2011).	82
6.3	Fluxo da personalização de dados contextuais (Miele <i>et al.</i> , 2009).	83

Lista de Tabelas

6.1	Classificação dos trabalhos relacionados.	84
6.2	Informações utilizadas para recomendação e personalização de conteúdo. .	85

Lista de Abreviaturas e Siglas

AEH: Adaptive Educational Hypermedia
AH: Adaptive Hypermedia
C&A: Captura e Acesso
CAA: Contextual Access Architecture
CAL: Content Abstraction Layer
CPref-SQL: Conditional Preference - SQL
CSS: Cascading Style Sheets
CX: Classroom eXperience
DER: Diagrama Entidade-Relacionamento
HTML: HyperText Markup Language
IDC: International Data Corporation
IHC: Interação Humano-Computador
LOM: Learning Object Metadata
NCL: Nested Context Language
NUI: Natural User Interface
P2P: Peer-to-peer
PCs: Personal Computers
RFID: Radio Frequency IDentification
SCORM: Sharable Content Object Reference Model
SQL: Structured Query Language
TV: Televisão
XML: eXtensible Markup Language
XSLT: eXtensible Stylesheet Language for Transformation

Capítulo 1

Introdução

1.1 Contextualização

O uso de dispositivos computacionais na vida das pessoas tornou-se uma tendência e, atualmente, está clara a sua difusão em diferentes domínios (Lee e Chen, 2009) como, por exemplo, em ambientes educacionais com salas de aula instrumentadas com projetores multimídia, lousas eletrônicas, microfones, câmeras de vídeo, dentre outros.

A presença de computadores tem se tornado cada vez mais intrínseca em nosso cotidiano (Weiser, 1991). De acordo com pesquisas de mercado realizadas pela empresa norte-americana *International Data Corporation (IDC)*, o número de *smartphones* e dispositivos móveis tem crescido a cada ano (IDC, 2013). Essa é uma tendência computacional cientificamente chamada de “*computação ubíqua*” e que traz às pessoas a possibilidade de lidar com os dispositivos de computação de forma mais habitual e não-invasiva.

A primeira utilização do termo computação ubíqua aconteceu no final dos anos 1980 por Weiser (1993) e, hoje, é uma área de pesquisa emergente (Lee e Chen, 2009) cujo principal objetivo é tornar invisível a interação entre humanos e computadores, ou seja, integrar a computação nas ações cotidianas das pessoas, fazendo com que elas executem tarefas com a ajuda de computadores, sem notá-los.

Essa é uma nova era computacional que surge e toma o lugar da era dos computadores pessoais na qual um computador era utilizado por apenas uma pessoa. Hoje, percebemos que estamos cercados de pequenos computadores e uma única pessoa utiliza inúmeros deles ao seu redor, como *notebooks*, *smartphones* e *tablets*.

Três grandes temas dividem a computação ubíqua: Interfaces Naturais, Ciência de Contexto e Captura e Acesso (C&A) (Abowd e Mynatt, 2000). O primeiro tema está relacionado com o suporte à linguagem natural e gestos na comunicação entre humanos e computadores para aproveitar mais das ações humanas implícitas no mundo. O segundo

tema, Ciência de Contexto, está ligado à ideia de os computadores poderem perceber e reagir ao ambiente em que estão inseridos, com o intuito de facilitar as atividades dos seres humanos. Por fim, C&A implica na tarefa de preservar a gravação de alguma experiência ao vivo para ser revisitada em algum ponto no futuro.

Em particular, sistemas automatizados que permitem a captura de experiências ao vivo fazem parte de um tema de pesquisa recorrente (Abowd e Mynatt, 2000; Truong e Hayes, 2009). Salas de aulas e de reuniões equipadas com lousas eletrônicas, microfones e câmeras de vídeo produzem artefatos que permitem reconstruir as experiências capturadas para posterior utilização e revisão, evitando que os usuários percam algum ponto importante enquanto fazem anotações, por exemplo. Sistemas dessa natureza auxiliam e melhoram a interatividade dentro das salas de aula e permitem uma maior eficiência do ensino (AiHua, 2010; Zeng *et al.*, 2010).

Esse processo de captura de atividades humanas pode ser automatizado através de aplicações de C&A que geram conteúdo multimídia em diversos formatos para que sejam posteriormente recuperados, visualizados e, porventura, estendidos ao longo do tempo.

Considerando a captura “ao vivo” das experiências cotidianas, o conteúdo gerado é caracterizado como multimídia (Moos e Marroquin, 2010), pois possui uma combinação de diferentes tipos de mídia, como por exemplo, vídeo, som e texto, levando à geração de um enorme volume de dados.

Dessa forma, é imprescindível que a recuperação e visualização do conteúdo produzido sejam claras, de fácil acesso e, principalmente, que estejam de acordo com as necessidades dos usuários, deixando-os satisfeitos quanto ao conteúdo apresentado, o que se torna um desafio para esse tipo de sistema computacional.

1.2 Motivação

Quando em sala de aula ou estudando, os alunos normalmente tendem a gastar uma parte significativa do seu tempo ouvindo, escrevendo e lembrando do conteúdo das aulas, geralmente verificando suas próprias anotações. Com isso, assuntos relacionados à gravação de aulas para posterior recuperação, autoria multimídia e melhoria do aprendizado por meio da Internet têm se tornado cada vez mais atrativos para escolas e universidades que visam melhorar a interatividade entre instrutores e alunos.

Nesses ambientes educacionais instrumentados, as aplicações de C&A utilizam mecanismos para gravação de diferentes tipos de mídia como áudio, vídeo e anotações feitas sobre lousas eletrônicas.

Com isso, nos deparamos com problemas relativos a altas demandas computacionais e de comunicação (Perich *et al.*, 2004). Implementações tradicionais encontradas na literatura apresentam problemas de escalabilidade causados por entidades centralizadas que se tornam gargalos (Hasan *et al.*, 2005), contrariando os requisitos de transparência de sis-

temas dessa natureza, propiciados pela alta disponibilidade (Ali Al-Dahoud e Walkowiak, 2010). O principal aspecto da alta disponibilidade neste contexto é a redundância de informações que devem ser transmitidas para diferentes pontos na rede através de canais de comunicação confiáveis.

Além disso, ainda lidamos com o desafio de apresentar todo esse conteúdo capturado de forma organizada e com interfaces intuitivas para auxiliar os alunos em seu processo de aprendizagem.

Assim, Hipermídia Adaptativa (Adaptive Hypermedia – AH) (Brusilovsky, 2001) surgiu como uma solução para o problema de personalização na Web, mais especificamente a Hipermídia Adaptativa Educacional (Adaptive Educational Hypermedia – AEH) (Brusilovsky, 2012) para atender as necessidades de cada estudante, adaptando o conteúdo aos seus objetivos, níveis de conhecimento, interesses, preferências e estilos de aprendizado. Os usuários desse tipo de sistema se esforçam para acessar as informações relevantes, procurando determinados trechos dos artefatos capturados com suas necessidades específicas e suas preferências em mente.

O conteúdo armazenado em um sistema automatizado de C&A pode ter pouco uso se não fornecer aos usuários os meios adequados para conseguir a informação almejada, o que inclui:

- Recomendação de conteúdo: são técnicas computacionais que possuem o objetivo de selecionar itens relevantes para cada usuário de acordo com seus interesses (Liang *et al.*, 2007);
- Recursos de personalização: ambientes personalizados de acordo com interesses e necessidades dos usuários baseado em suas informações pessoais e contexto de acesso (Liang *et al.*, 2007);
- Suporte a preferências do usuário: mecanismos para coleta de informações dos usuários que permitem identificar suas preferências e, com isso, sugerir itens relevantes individualmente (de Amo e Ribeiro, 2009; Liang *et al.*, 2007).

Um bom exemplo de tais demandas é o caso de um estudante que deseja acessar o conteúdo relacionado à próxima prova de uma disciplina. Baseado em seu contexto durante o acesso (por exemplo, localização, tempo disponível para estudo, recursos do dispositivo, etc.), o sistema poderia proporcionar-lhe uma experiência de acesso personalizado, provendo uma interface que aponta diretamente para o conteúdo da próxima prova agendada (por exemplo, aulas 2 a 6 são conteúdos para essa prova).

A personalização poderia ainda ir além e, por exemplo, adaptar os conteúdos já filtrados para restrições mais específicas, por exemplo, excluir qualquer conteúdo de áudio porque fones de ouvido não estão disponíveis no dispositivo do usuário, ou ainda, mostrar somente os slides da aula ministrada, sem o vídeo nem o áudio, porque a largura de banda disponível para o dispositivo é baixa.

Visando contribuir com soluções dessa natureza, o trabalho aqui proposto utiliza conceitos de dois temas da computação ubíqua, Ciência de Contexto e Captura e Acesso, aplicados ao domínio educacional.

1.3 Objetivos

Este trabalho abrange, em largura, diversos aspectos de projeto e implementação de aplicações de C&A, principalmente (1) o armazenamento e distribuição de conteúdo multimídia e (2) o acesso contextualizado e personalização do conteúdo capturado.

Assim, primeiramente é apresentada um modelo baseado em redes *peer-to-peer* (P2P) para armazenamento e distribuição do conteúdo multimídia gravado para solucionar problemas relacionados ao surgimento de eventuais gargalos no meio de comunicação e transmissão do conteúdo capturado.

Além disso, foi elaborada uma arquitetura computacional para dar suporte ao acesso contextualizado e personalizado ao conteúdo multimídia capturado em ambientes educacionais instrumentados, de modo a atender as expectativas e necessidades de alunos, auxiliando-os na recuperação do conteúdo desejado.

É realizado um estudo de caso, aplicando a arquitetura proposta no sistema *Classroom eXperience* (Ferreira *et al.*, 2012a; Ferreira, 2012), que é uma plataforma de computação ubíqua para captura, armazenamento, acesso e extensão de informações multimídia referentes a atividades educacionais.

Dessa forma, espera-se formalizar uma arquitetura computacional, a partir de componentes de software fracamente acoplados, que poderá ser aplicada na fase de acesso à informação para a personalização de conteúdo multimídia em sistemas de C&A e que também ofereça suporte a preferências de usuários levando em consideração seu contexto de acesso.

Por fim, as *interfaces* de acesso contextual do *Classroom eXperience* foram projetadas de modo a se adequarem à nova experiência de acesso criada e levando em consideração os princípios, padrões e *guidelines* que são três pilares que apoiam os projetos de usabilidade em Interação Humano-Computador (IHC) (Dix *et al.*, 2004).

1.4 Método de Pesquisa

A pesquisa na área de computação ubíqua é caracterizada pela necessidade de realização prévia de estudos de caso e análises qualitativas e, por isso, dá-se um enfoque experimental (Truong, 2005; Weiser, 1993). Dessa maneira, esta pesquisa foi desenvolvida sobre uma abordagem exploratória (Wazlawick, 2009) com o intuito de extrair abstrações que pudessem compor a arquitetura proposta.

A utilização dessa metodologia de pesquisa tem sido confirmada na prática e a experiência tem demonstrado que, quando o progresso em uma área da computação ubíqua é atingido por meio da experimentação com protótipos de aplicações, deve-se começar a investigar as abstrações de projeto relevantes para a classe de aplicações em questão e então desenvolver e implementar uma solução arquitetural que contemple tais abstrações (Dey, 2001; Modahl *et al.*, 2004; Truong, 2005).

Protótipos funcionais são utilizados com o intuito de verificar a viabilidade de soluções propostas sem que haja a necessidade de criar a infraestrutura completa. Dessa forma, a solução proposta foi verificada através da construção de um protótipo considerando alguns desafios arquiteturais e de projeto de aplicações ubíquas focadas em usuários (Tang *et al.*, 2011). Em complemento, análises de usabilidade foram realizadas para identificar possíveis falhas nas interfaces e suas devidas correções foram propostas a fim de melhorar a experiência dos usuários.

1.5 Estrutura da Dissertação

Esta dissertação está organizada da seguinte maneira: o Capítulo 2 apresenta os conceitos sobre Computação Ubíqua, Captura e Acesso de Conteúdo Multimídia, Personalização de Conteúdo e Ciência de Contexto com o objetivo de dar um embasamento teórico para a compreensão do trabalho; o Capítulo 3 formaliza o modelo de comunicação proposto para armazenamento e distribuição de conteúdo multimídia; o Capítulo 4 apresenta a arquitetura desenvolvida para o acesso contextualizado e personalizado de conteúdo multimídia; no Capítulo 5, o *Classroom eXperience* é apresentado em detalhes como um estudo de caso; no Capítulo 6, alguns trabalhos encontrados na literatura são discutidos e relacionados com este; e, por fim, o Capítulo 7 apresenta as considerações finais sobre este trabalho, juntamente com suas limitações e trabalhos futuros.

Capítulo 2

Ambientes Educacionais Ubíquos: Fundamentos e Conceitos

A utilização de recursos computacionais e tecnológicos vem sendo amplamente utilizados e difundidos em diversos domínios. À medida que avançamos em direção a um ambiente de computação mais onipresente, nos aproximamos do que chamamos de computação ubíqua, na qual a presença de computadores se torna menos visível e, eventualmente, eles se misturam em nossas vidas cotidianas.

As pessoas podem aprender e/ou trabalhar em seu cotidiano a qualquer hora e em qualquer lugar usando seus dispositivos móveis interligados através de conexões de rede sem fio. Em particular, abordagens que auxiliam o processo de ensino e aprendizado estão em ascensão.

Os computadores estão cada vez mais integrados nas salas de aulas. *Notebooks, tablets, smartphones* e vários outros dispositivos estão sendo utilizados no apoio aos instrutores no ensino e também aos alunos para absorverem os conhecimentos transmitidos pelos instrutores. Especialmente em escolas e universidades, estudantes podem acessar o conteúdo das aulas e atividades de qualquer lugar e a qualquer momento. Essa abordagem é chamada de *ubiquitous learning (u-Learning)* (Zhao *et al.*, 2010).

A captura “ao vivo” de experiências cotidianas gera um grande volume de conteúdo digital em diversos formatos. Por isso, esse conteúdo é caracterizado como multimídia (Moos e Marroquin, 2010), pois possui uma combinação de diferentes tipos de mídia como, por exemplo, vídeo, som e texto.

Neste capítulo são apresentados os principais conceitos relacionados a este trabalho. Na Seção 2.1, a computação ubíqua e suas sub-áreas são apresentadas, pois são os pilares deste trabalho; a Seção 2.2 apresenta a computação ubíqua especificamente incorporada ao domínio educacional; e, por fim, as considerações finais são feitas na Seção 2.3.

2.1 Computação Ubíqua

Os computadores, na época de seu surgimento, eram utilizados principalmente por corporações e organizações governamentais e eram tão grandes que ocupavam enormes salas. Esses computadores eram chamados de *mainframes*. Nessa época, um único computador era utilizado por várias pessoas.

Com o passar dos anos, o tamanho dos equipamentos computacionais têm diminuído. Assim, surgiu a era dos computadores pessoais, ou *Personal Computers (PCs)*, na qual o tamanho, a capacidade e o preço de venda torna os computadores viáveis para cada pessoa individualmente, fazendo com que um único computador seja utilizado por apenas uma pessoa.

Com a evolução da Internet, da computação distribuída e dos dispositivos móveis que estão cada vez menores, surge uma nova era, onde os computadores pessoais dão lugar à computação ubíqua, onde uma única pessoa utiliza vários computadores (Weiser e Brown, 1996). Um infinidade de dispositivos fazem parte da computação ubíqua. Alguns deles são mostrados na Figura 2.1.



Figura 2.1: Alguns exemplos de dispositivos de computação ubíqua.

A computação ubíqua pode ser definida como uma área da ciência da computação cuja principal característica é a inserção de recursos computacionais de qualquer natureza intrinsecamente no ambiente, com o intuito de auxiliar os seres humanos a realizarem suas tarefas cotidianas (Weiser, 1993).

Desta forma, a computação ubíqua é uma área de pesquisa multidisciplinar que lida com Computação Móvel, Sistemas Distribuídos, Interação Humano-Computador, Recuperação de Informação, entre outras de suas disciplinas. Isso leva a avanços científicos na computação, proporcionados pela criação de novos problemas e soluções computacionais reais.

Hoje é possível perceber que a computação ubíqua está espalhada e intrínseca à sociedade. O conceito de cidades inteligentes, por exemplo, surgiu da necessidade de gerir

os problemas causados pelo crescimento descontrolado da população nos grandes centros urbanos (da Silva *et al.*, 2013). A ideia principal é que dispositivos sejam espalhados nas cidades para que dados sejam coletados e analisados com o intuito de garantir a prestação de serviços em níveis satisfatórios de qualidade e efetividade.

Em uma visão mais ampla, a computação ubíqua pode ser dividida em três grandes temas de pesquisa (Abowd e Mynatt, 2000): Interfaces Naturais, Ciência de Contexto e Captura e Acesso (C&A).

O primeiro deles, chamado de Interfaces Naturais (*Natural User Interfaces* ou NUI, em inglês), estuda a comunicação entre humanos e computadores através da linguagem natural dos humanos, como reconhecimento de voz, de gestos, de toques, entre outros. Existem estudos que correlacionam as tecnologias de interação adequadas a diferentes situações (Wigdor e Wixon, 2011). O crescimento dessa área acontece principalmente através dos dispositivos de videogame que estudam e exploram diferentes formas de interação com o intuito de tornar o jogo mais real, como por exemplo, *Nintendo Wii* e do *XBox 360 + Kinect*¹.

O segundo tema, Ciência de Contexto (*Context-awareness*, em inglês), está ligado à ideia de os computadores poderem perceber e reagir ao ambiente em que estão inseridos visando facilitar as atividades dos seres humanos.

E por fim, C&A que é responsável por registrar as experiências vividas no dia-a-dia das pessoas e desejam revisitá-las futuramente.

Sendo assim, o trabalho aqui proposto utiliza recursos e conceitos dos dois últimos temas apresentados, Captura e Acesso e Ciência de Contexto, os quais serão detalhados a seguir.

2.1.1 Captura e Acesso

C&A é a sub-área da computação ubíqua que trata da captura de atividades para que possam ser recuperadas posteriormente, ou seja, aplicações que permitem que atividades cotidianas sejam gravadas, gerando artefatos que possam ser posteriormente acessados pelas pessoas (Truong, 2005).

As aplicações de C&A fornecem recursos especializados para gravação ao vivo de eventos, garantindo aos usuários que os detalhes da experiência vivida possam ser revisitados no futuro.

Porém, as atividades de captura e acesso não são apenas aquelas que gravam e disponibilizam as informações, mas sim, várias outras atividades que ocorrem entre essas etapas e também após elas. A etapa de armazenamento, por exemplo, deve prover meios eficientes de transmissão e armazenamento de todo o conteúdo para facilitar sua posterior recuperação. Após serem armazenados, os conteúdos podem ainda ser enriquecidos com

¹<http://www.xbox.com/pt-BR/Kinect>

novas informações ou, ainda, ser associados à outros conteúdos relacionados através de ligações hipermídia.

As aplicações de C&A podem estar contidas em diversos domínios de aplicação e áreas de conhecimento, cada um com suas particularidades e características. Porém, o trabalho aqui proposto está focado em ambientes educacionais instrumentados, nos quais as aulas ministradas pelos instrutores são gravadas em salas de aula equipadas com projetores multimídia, lousa eletrônica, microfones e câmeras de vídeo para posterior recuperação do conteúdo pelos alunos.

2.1.2 Características das Aplicações de Captura e Acesso

Uma aplicação de C&A pode existir desde a forma mais simples, por meio de um único dispositivo para captura e acesso ou de uma forma mais complexa, com um conjunto de dispositivos que realizam a captura e o acesso.

Para que a arquitetura das aplicações de C&A seja padronizada, geralmente essas aplicações são estruturadas de acordo com as quatro fases propostas por Abowd *et al.* (1996):

- *Pré-produção*: fase de preparação do conteúdo que será capturado;
- *Gravação ao vivo*: consiste no processo de gravação que ocorre dentro do ambiente instrumentado;
- *Pós-produção*: processamento inicial do conteúdo capturado no momento em que a gravação acaba de ser realizada; e,
- *Acesso*: apresentação do conteúdo capturado ao usuário final.

Essas fases não são apenas simples passos que devem ser considerados na construção de aplicações de C&A. Na verdade, elas mostram como estruturar e organizar inteligentemente o papel de cada etapa de interação entre o usuário e o sistema.

Essa estrutura pode conter pequenas adaptações, como na proposta de Richter *et al.* (2001) na qual a segunda e a terceira fase são unidas em uma só. Pimentel *et al.* (2001) propõem ainda uma quinta fase, chamada de *extensão*, permitindo que os usuários complementem o processo de captura através da interação com o conteúdo capturado e visualizado.

Segundo Truong e Hayes (2009), cada uma dessas fases é responsável por incorporar quatro requisitos não-funcionais básicos das aplicações de C&A:

1. **A fase de captura deve acontecer naturalmente**: as aplicações dessa natureza devem auxiliar as pessoas a realizar suas tarefas diárias sem desviar sua atenção, por meio de um processo de captura que ocorra de forma discreta e exija pouco ou nenhum esforço adicional do usuário, ou seja, o ambiente deve ser preparado de

forma não intrusiva deixando seus componentes quase ou totalmente impercebíveis aos usuários;

2. **As informações capturadas devem ser acessíveis com o mínimo esforço:** a utilidade de aplicações dessa natureza se torna evidente na fase de acesso, onde os usuários revisam as atividades previamente capturadas. Com isso, a fase de acesso deve ser projetada para ser agradável aos usuários e as informações devem estar acessíveis por toda parte e a qualquer momento, atendendo suas necessidades;
3. **As informações gravadas devem ser indexadas pelo conteúdo e pelo contexto:** o contínuo uso dessas aplicações gera um enorme volume de dados que pode interferir diretamente na recuperação do conteúdo capturado. Contudo, essas aplicações devem oferecer aos usuários formas claras para acessar esse conteúdo. Para isso, índices como títulos e palavras-chave devem ser criados para que os usuários encontrem facilmente o conteúdo almejado; e, por último,
4. **Os registros devem ser criados, armazenados e acessados de modo ético e legal:** as aplicações devem garantir as proteções adequadas contra a utilização indevida do conteúdo capturado, bem como os desafios sociais relacionados através da classificação das informações em públicas e privadas, além da criação de uma política de privacidade.

Esses requisitos foram observados a partir de uma revisão de aplicações bem sucedidas que utilizam tecnologias de computação ubíqua para o problema de grande escala e gravação “ao vivo” de forma natural (Truong e Hayes, 2009).

2.1.3 Ciência de Contexto

Quando seres humanos conversam uns com os outros, eles são capazes de usar as informações situacionais (também chamadas de contexto) que estão implícitas no ambiente para melhorar e entender a conversação. No entanto, essa capacidade não acontece tão naturalmente quando computadores estão envolvidos nas interações.

Assim, a ciência de contexto, ou *context-awareness* em inglês, é uma sub-área da computação ubíqua na qual os computadores são capazes de perceber o contexto em que estão inseridos com o intuito de prover informações relevantes para ajudar os usuários a executarem suas tarefas.

Em 1994, Schilit e Theimer (1994) discutiram “computação ciente de contexto” pela primeira vez como sendo “aplicações que se adaptam de acordo com a localização de seu uso, o conjunto de pessoas e objetos próximos, bem como as mudanças ocorridas nesses objetos ao longo do tempo”.

Várias outras definições de ciência de contexto foram criadas por diferentes autores e podem ser encontradas na literatura. No entanto, todas elas remetem a dois principais

aspectos: a utilização de contexto e a adaptação a esse contexto (Dey, 2001). O objetivo principal é melhorar a riqueza da comunicação nas interações entre humanos e computadores através da melhoria dos aspectos contextuais disponibilizados aos computadores, fazendo com que mais informações e serviços computacionais mais úteis sejam gerados.

Segundo os estudos de Abowd *et al.* (1999), as aplicações orientadas à contexto podem ser caracterizadas em três grupos:

- **Apresentação de informações e serviços aos usuários:** é o grupo de aplicações que recuperam as informações de contexto e executam comandos de forma manual, ou seja, as informações são solicitadas aos usuários;
- **Execução automática de um serviço:** são aplicações capazes de executar ou adaptar os serviços providos automaticamente de acordo com as informações de contexto; e,
- **Marcação de contexto nas informações para posterior recuperação:** são aplicações capazes de associar informações digitais com o contexto do usuário para, por exemplo, permitir a visualização de informação caso o usuário esteja em determinado contexto.

A Figura 2.2 mostra uma pessoa obtendo informações contextualizadas sobre coisas, pessoas e acontecimentos perto do local onde ela está, através de seu dispositivo móvel.



Figura 2.2: Obtendo informações contextuais através de um dispositivo móvel (Business Computing World, 2011).

2.2 Ambientes Educacionais Ubíquos

Como já detalhado na seção anterior, ambientes de computação ubíqua permitem que qualquer pessoa possa fazer uso de computadores que estão integrados em todos os lugares e a qualquer momento.

No domínio educacional não é diferente. Salas de aula estão sendo equipadas com diversos dispositivos computacionais, como projetores multimídia, lousas eletrônicas, microfones, câmeras de vídeo, *tablets*, sensores, dentre outros.

Desta forma, as técnicas de computação ubíqua têm atraído a atenção de escolas e universidades com o intuito de dinamizar e tornar o processo de ensino e aprendizado mais interativos. E assim, os alunos são incentivados a participar colaborativamente das aulas, oferecendo recursos de aprendizagem antes inimagináveis.

O registro de informações em sala de aula não é uma tarefa nova. Por exemplo, o *eClass* (também conhecido como *Classroom 2000*) é um projeto iniciado em 1996 com o objetivo de gravar informações das aulas ministradas por instrutores (Abowd, 1999; Brotherton e Abowd, 2004). Além disso, as informações previamente capturadas são disponibilizadas posteriormente aos alunos via *Web*.

Ainda em 1999, o sistema *Student Notepad (StuPad)* (Truong *et al.*, 1999) foi integrado ao *eClass* para permitir que os alunos pudessem visualizar, em seus computadores pessoais, o que o instrutor ministra.

Assim, a computação ubíqua aplicada na educação é um tema de pesquisa emergente e pode ser chamado de “aprendizagem ubíqua”, ou *ubiquitous learning (u-learning)*, em inglês. *U-learning* é uma abordagem que combina ambientes de aprendizagem reais e virtuais que, juntos, geram artefatos de estudo mais ricos.

Os resultados dos estudos realizados por Settle *et al.* (2011) mostram que a utilização de tecnologias que registram o conteúdo das aulas ministradas pelos instrutores tem um impacto positivo no aprendizado dos alunos.

Os instrutores devem estar preparados para ensinar nas escolas que estão integradas em um mundo onde a tecnologia, particularmente as tecnologias digitais portáteis, estão mudando a forma pela qual nós vemos e damos sentido às coisas, bem como interagimos uns com os outros (Chien, 2012).

2.3 Considerações Finais

Neste capítulo foram apresentados alguns conceitos fundamentais para a compreensão do trabalho proposto. Como se pode perceber, temas como Computação Ubíqua, Captura & Acesso e Ciência de Contexto estão cada vez mais presentes no dia-a-dia das pessoas, auxiliando-as na execução de suas tarefas diárias.

Em especial, ambientes educacionais instrumentados ajudam tanto alunos em seu

processo de aprendizado, quanto instrutores em seu processo de ensino, tornando suas experiências mais interativas, em sala de aula e fora dela. Para isso, as aplicações de C&A, particularmente aquelas cientes de contexto, têm fundamental importância durante esse processo.

Capítulo 3

Um Modelo para Distribuição de Conteúdo Multimídia Usando Redes *Peer-to-Peer*

Em ambientes instrumentados para captura multimídia, enfrentam-se problemas relacionados a grandes demandas computacionais e de comunicação. Um dos grandes desafios é disponibilizar o conteúdo capturado pelas aplicações de C&A aos usuários, deixando-o sempre disponível para que seja acessado a qualquer hora e em qualquer lugar.

Implementações tradicionais encontradas na literatura apresentam problemas de escalabilidade e alta disponibilidade causados por entidades centralizadas que tornam-se, frequentemente, gargalos do sistema (Hasan *et al.*, 2005). Questões como acesso remoto, alta disponibilidade, gerenciamento de energia e acesso à informação em dispositivos móveis têm aumentado e, em paralelo, a computação ubíqua tira proveito da computação distribuída e móvel (Cook e Das, 2012).

Dessa forma, os sistemas devem oferecer alta disponibilidade para garantir os requisitos de transparência dos serviços para os usuários. Nesse contexto, o principal aspecto para garantir a alta disponibilidade é a redundância de informações, as quais devem ser transmitidas para diferentes pontos da rede por meio de canais de comunicação confiáveis (Kanso *et al.*, 2011).

Nishiuchi *et al.* (2010), Kunz *et al.* (2010) e Giliberti *et al.* (2011) utilizam uma arquitetura cliente/servidor em suas abordagens para transmitir o conteúdo capturado para servidores responsáveis pelo seu armazenamento. Porém, como já mencionado anteriormente, essa abordagem pode se tornar um gargalo.

Outras abordagens utilizam arquiteturas *peer-to-peer* (P2P) (Barolli e Xhafa, 2011; Kameas *et al.*, 2002; Xie *et al.*, 2010) com o intuito de resolver o problema citado, em que

um computador pode exercer papéis tanto de cliente como de servidor. O trabalho ora proposto segue essa vertente ao formalizar e implementar uma camada de abstração para transferência e replicação de conteúdo para aproveitar os benefícios que arquiteturas P2P podem oferecer, como escalabilidade, tolerância a falhas e compartilhamento de recursos (Lin *et al.*, 2007).

Neste capítulo, tal camada de abstração é apresentada em detalhes. A Seção 3.1 descreve a arquitetura e os aspectos gerais dessa camada; na Seção 3.2, o protocolo de comunicação utilizado é especificado e formalizado com a apresentação de seus aspectos de projeto; e, por último, na Seção 3.3 são feitas as considerações finais.

3.1 Camada de Abstração para Transferência e Replicação de Conteúdo

Com o objetivo de evitar os problemas ocorridos em abordagens que utilizam arquitetura cliente/servidor, uma camada de comunicação foi projetada sobre uma arquitetura *peer-to-peer*. Essa camada foi nomeada de *Content Abstraction Layer* (CAL) (Araújo *et al.*, 2012).

A camada CAL visa abstrair os serviços de transferência e replicação de conteúdo multimídia capturado através de aplicações de C&A em ambientes ubíquos. Utilizando essa camada, as aplicações não precisam saber em qual ou quais computadores o conteúdo está armazenado, ou seja, o local onde o conteúdo é armazenado é transparente à elas. Além disso, a CAL é responsável por replicar todo esse conteúdo que já foi armazenado entre os *peers* da rede, visando aumentar a disponibilidade desse conteúdo ao recuperá-lo.

A camada CAL foi construída sobre a plataforma JXTA (Sun Microsystems, 2007) com o intuito de utilizar seus serviços para abstrair o tráfego do conteúdo através das camadas inferiores, ou seja, a transferência de conteúdo se torna transparente para as aplicações que a utilizam.

A Figura 3.1 mostra que a camada CAL está localizada entre a aplicação P2P e a plataforma JXTA.

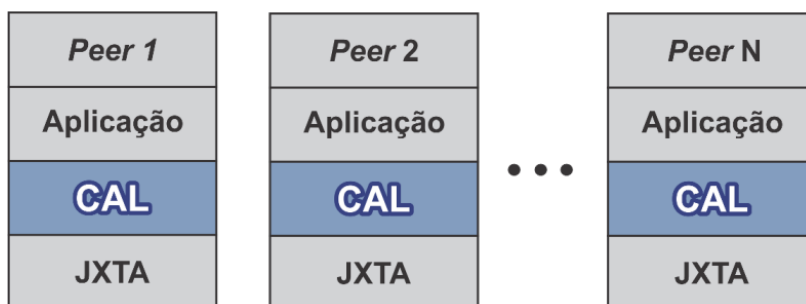


Figura 3.1: Localização da camada CAL.

A plataforma JXTA¹ foi escolhida e estudada principalmente por ser de código aberto, por sua estabilidade e também por sua grande utilização pela comunidade acadêmica (Barolli e Xhafa, 2011; Zhong-xian *et al.*, 2010).

Tal plataforma fornece um conjunto de mecanismos simples, pequenos e flexíveis que suportam a comunicação P2P em qualquer plataforma operacional. Usando esses mecanismos é possível que os dispositivos integrados em uma rede privada possam se conectar a *peers* na Internet, passando por barreiras de *firewalls* e NATs (Network Address Translation) (Barolli e Xhafa, 2011).

Cada um desses *peers* podem ser diferentes dispositivos de rede, como sensores, celulares, computadores pessoais, *tablets*, entre outros. Esses *peers* podem ser organizados em grupos para fazer uso de serviços comuns entre eles, mas todos os integrantes do grupo devem implementar esses serviços. Cada grupo pode determinar suas próprias políticas de membros desde *aberto* até *altamente seguro e protegido*. Além disso, cada *peer* pode pertencer a mais de um grupo simultaneamente.

Um conjunto de serviços básicos é definido pelo JXTA. Dentre eles, o serviço de descoberta (chamado de *discovery*) é utilizado pelos membros dos grupos para a busca de recursos, tais como: serviços, *pipes*, *peers* e grupos. As conexões de *pipes* entre *peers* membros de um grupo são criadas e gerenciadas pelo serviço *pipe service*. Em síntese, *pipes* são mecanismos síncronos e unidirecionais para transferência de mensagens entre *peers*. Mesmo que os dispositivos não tenham uma conexão física direta, os *pipes* criam uma conexão lógica e, por isso, são tratados como canais de comunicação virtual.

3.2 Especificação e Projeto do Protocolo de Comunicação

A camada CAL utiliza um protocolo de comunicação próprio que foi desenvolvido no âmbito desta dissertação. Os cinco elementos essenciais de um protocolo de redes de computadores, como ambiente, vocabulário, formato, serviços e regras procedimentais (Holzmann, 1991), são descritos nesta seção.

O primeiro deles, chamado de ambiente, define qual será o ambiente em que o protocolo será utilizado, levando em consideração suas características físicas e lógicas, como arquitetura, organização dos computadores, etc. O vocabulário é o conjunto de nomes das mensagens utilizadas para especificar as transições dos estados do protocolo e o formato que, como o nome já diz, é o formato de cada uma dessas mensagens. Os serviços são elementos abstratos que definem as funcionalidades disponíveis e o comportamento de tais funcionalidades é definido por regras procedimentais através de autômatos.

¹<https://java.net/projects/jxta>

3.2.1 Ambiente

O protocolo atua exclusivamente em redes *peer-to-peer* sobre a plataforma JXTA. Cada *peer* na rede pode atuar tanto no papel de produtor quanto de consumidor de conteúdo.

Enquanto produtor de conteúdo, o *peer* é responsável pela captura do conteúdo do mundo real através de dispositivos instalados no ambiente e produz artefatos multimídia referentes a essas sessões de captura.

Ao atuar como consumidor, o *peer* recebe e armazena o conteúdo enviado pelos *peers* produtores. Além disso, caso esse *peer* atue nos dois papéis, ele também pode compartilhar seu conteúdo com os outros *peers* consumidores na rede com o intuito de prover redundância e alta disponibilidade do conteúdo aos usuários.

A Figura 3.2 mostra uma topologia de rede com *peers* consumidores e produtores. Os *peers* P1, P2 e P3 atuam na rede como *peers* produtores, o *peer* P4 é tanto produtor quanto consumidor pois provê serviço de armazenamento e os *peers* P5 e P6 são somente consumidores e armazenam o conteúdo produzido pelos *peers* produtores.

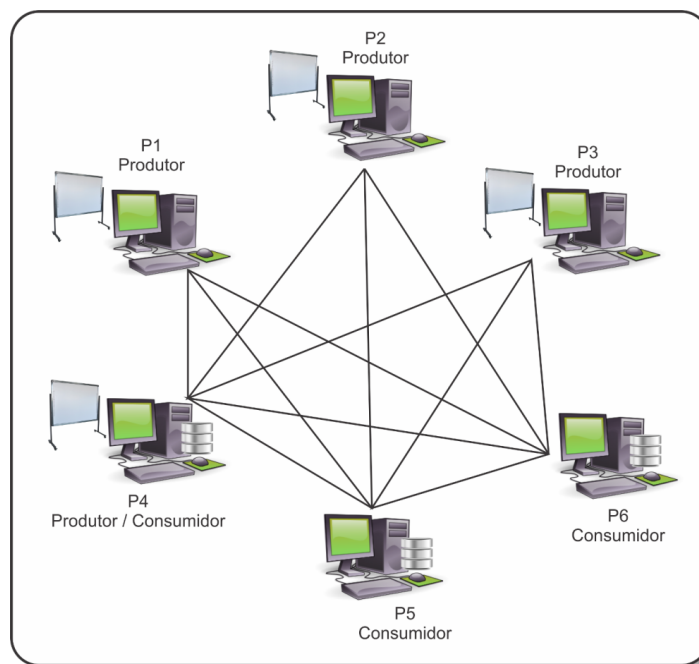


Figura 3.2: Topologia de rede com *peers* produtores e consumidores.

3.2.2 Vocabulário

O vocabulário é o conjunto das mensagens conhecidas pelo protocolo que são utilizadas durante a comunicação. O conjunto de mensagens desse protocolo pode ser dividido em dois grupos: um conjunto das mensagens que são trocadas entre a aplicação P2P e a camada CAL e um outro conjunto das mensagens que são trocadas entre a camada CAL e a plataforma JXTA.

O conjunto de mensagens trocadas entre a aplicação P2P e a camada CAL é:

- *start_session_req*: qualquer operação realizada entre dois *peers* deve acontecer dentro de uma sessão. Para iniciar uma sessão, essa mensagem é enviada à camada CAL pela aplicação P2P para solicitar a abertura de uma sessão entre dois *peers*;
- *start_session_conf+*: essa mensagem indica que o início da sessão ocorreu sem nenhuma falha;
- *start_session_conf-*: essa mensagem é enviada quando há alguma falha no reconhecimento da mensagem *start_session_req*;
- *end_session_req*: para encerrar uma conexão, a aplicação P2P envia essa mensagem para a camada CAL que, em sequência, envia a mensagem *disconnect_req* para concluir a operação;
- *ls_req* (*ls* é uma abreviação para *list_status*): essa mensagem é utilizada apenas entre dois *peers* consumidores e serve para consultar a lista de conteúdo (lista de arquivos) disponíveis em outro *peer*. Essa é a primeira etapa para dois *peers* replicarem seus conteúdos;
- *ls_conf+*: ao receber uma mensagem *ls_request* válida, a camada CAL retorna essa mensagem para indicar seu reconhecimento;
- *ls_conf-*: mensagem enviada pela camada CAL caso a mensagem *ls_request* não seja reconhecida devido a alguma falha;
- *ft_get_req* (*ft* é uma abreviação para *file_transfer*): essa mensagem é enviada somente por *peers* consumidores para requisitar um conteúdo específico de outro *peer*;
- *ft_get_conf+*: mensagem usada para indicar que o último *ft_get_req* foi reconhecido corretamente;
- *ft_get_conf-*: a camada CAL utiliza essa mensagem para sinalizar a aplicação P2P que houve falha no reconhecimento da mensagem *ft_get_req*;
- *ft_put_req*: essa mensagem é enviada apenas por *peers* produtores que desejam enviar o conteúdo capturado. Assim, essa mensagem é utilizada para sinalizar o envio de um conteúdo;
- *ft_put_conf+*: essa mensagem é utilizada para indicar que o último *ft_put_req* foi reconhecido;
- *ft_put_conf-*: essa mensagem sinaliza a aplicação P2P que o último *ft_put_req* não foi reconhecido;
- *send_seg_req*: como o conteúdo a ser enviado é geralmente muito grande, ele é dividido em partes menores. Assim, essa mensagem é utilizada para enviar apenas um segmento desse conteúdo. Essa mensagem pode ser enviada tanto pelo produtor quanto pelo consumidor;

- *send_seg_resp*:- quando uma mensagem de envio de segmento (*send_seg_request*) não é reconhecida ou possui alguma falha, então a camada CAL retorna a mensagem *send_seg_resp*-. Perceba que a mensagem positiva *send_seg_response*+ não é utilizada, pois essa mensagem é utilizada por um serviço com reconhecimento negativo (Holzmann, 1991).

O conjunto de mensagens trocadas entre a camada CAL e a plataforma JXTA:

- *connect_req*: a conexão com outro *peer* na rede é solicitada a partir do envio dessa mensagem pela camada CAL;
- *connect_conf*+: essa mensagem é utilizada para sinalizar a camada CAL que a conexão foi bem-sucedida;
- *connect_conf*:- essa mensagem é utilizada para sinalizar a camada CAL que houve algum erro na conexão;
- *discovery_req*: após a aplicação P2P solicitar a abertura de uma sessão entre os *peers*, a camada CAL envia essa mensagem para procurar por *peers* consumidores disponíveis na rede;
- *discovery_conf*+: essa mensagem indica que algum *peer* consumidor foi encontrado na rede;
- *discovery_conf*:- essa mensagem indica que nenhum *peer* consumidor foi encontrado na rede;
- *disconnect_req*: essa mensagem é usada pela camada CAL para encerrar uma conexão entre dois *peers*;
- *send_msg_req*: essa mensagem é utilizada pela camada CAL para encapsular os serviços requisitados pela aplicação P2P e enviá-los através do JXTA.

3.2.3 Formato

Todas as mensagens do vocabulário desse protocolo são codificadas por meio do método orientado a caracteres (Holzmann, 1991), ou seja, os pacotes transmitidos são montados com a utilização de caracteres.

Além disso, todas as mensagens possuem o mesmo formato com *header*, *body* e *trailer*, conforme mostrado na Figura 3.3. O *header* é formado por três campos:

1. O campo *MORE_BIT* que é utilizado para indicar se existem ou não mais segmentos de dados a serem transferidos. Essa situação ocorre quando um *peer* consumidor deseja efetuar uma replicação de conteúdo com um segundo *peer* que possui uma grande quantidade de conteúdo que excede o tamanho da primitiva ou quando um *peer* produtor precisa enviar o conteúdo capturado e esse conteúdo é tão grande que precisa ser quebrado em partes menores.

2. O campo *SERVICE* representa o nome do serviço desejado e é codificado em um número binário. Como a camada CAL possui oito serviços disponíveis, três bits são suficientes para representar todos os serviços ($2^3 = 8$).
3. O campo *CONTENT_LENGTH* armazena o tamanho do conteúdo que está sendo enviado no campo de dados da mensagem, permitindo que ele seja variável.

O campo *body* contém o conteúdo em si a ser transferido. O tamanho máximo desse campo é 64Kbytes.

Por fim, o campo *trailer* contém o código *hash* da mensagem utilizado para verificação de erro. Esse código é encriptado utilizando o algoritmo MD5 e, por isso, são necessários 16bytes para seu armazenamento.

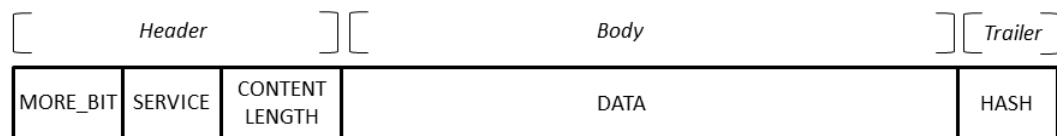


Figura 3.3: Formato de primitiva do protocolo proposto.

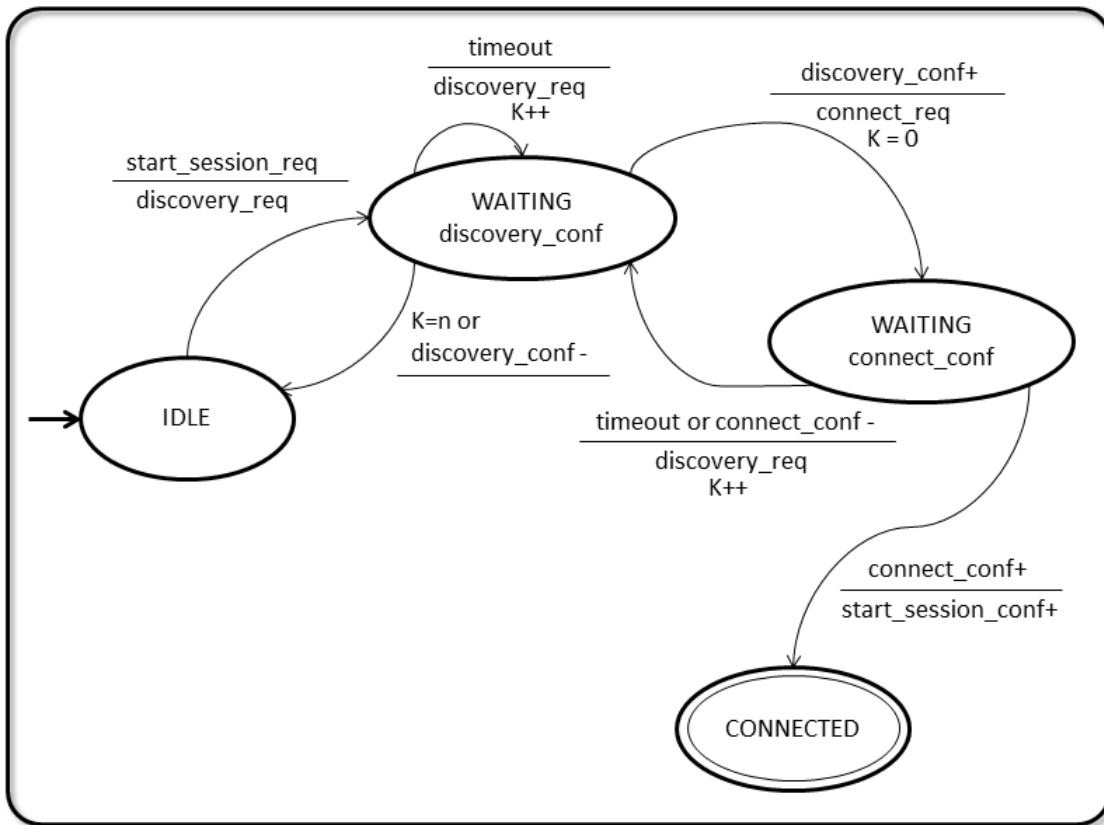
3.2.4 Serviços e Regras Procedimentais

O protocolo proposto apresenta oito serviços que são disponibilizados através da camada CAL. São eles: *publish*, *start_session*, *end_session*, *list_status*, *ft_get*, *ft_put*, *send_seg* e *reject*.

O primeiro deles, chamado de *publish*, é utilizado quando algum *peer* da rede deseja tornar-se um consumidor, ou seja, esse *peer* deseja disponibilizar seu espaço em disco para armazenar conteúdo de outros *peers*. Esse serviço é não confirmado, ou seja, o *peer* envia essa primitiva e não aguarda por confirmação.

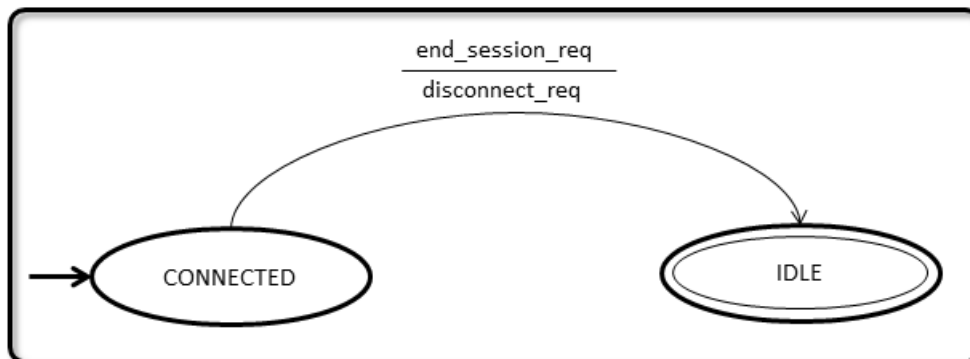
O serviço *start_session* representa a fase de estabelecimento de uma sessão de comunicação entre dois *peers*. Esse serviço é obrigatório para todos os *peers* que desejam se comunicar com outro *peer*, independentemente do papel que eles exercem (produtor ou consumidor). A Figura 3.4 mostra o autômato que representa o comportamento desse serviço.

Para iniciar uma sessão com outro *peer*, a aplicação P2P envia uma requisição de *start_session* para a camada CAL enviando a mensagem *start_session_req*. Nesse momento, a camada CAL chama o serviço *discovery* da plataforma JXTA para encontrar um *peer* disponível na rede e sai do estado IDLE e espera a mensagem *discovery_conf* com as informações necessárias para estabelecer a conexão com o outro *peer*. Se acontecer um *timeout*, a camada tenta novamente até que o limite n (previamente definido) seja atingido. Caso receba a mensagem *discovery_conf+*, o serviço *connect* da plataforma JXTA é invocado e, então, espera uma confirmação que pode ser positiva ou negativa.

Figura 3.4: Autômato do serviço *start_session*.

Se positiva, a camada vai para o estado **CONNECTED** e sinaliza a aplicação através da mensagem *start_session_conf+*. Se negativa ou *timeout*, a camada tenta encontrar um outro *peer* disponível até que o número pré-estabelecido de tentativas seja atingido.

Para encerrar uma sessão de comunicação com outro *peer*, a aplicação chama o serviço *end_session* da camada CAL. Ao receber a mensagem *end_session_req*, a camada chama o serviço *disconnect* da plataforma JXTA e, por ser um serviço não confirmado, volta automaticamente para o estado **IDLE**, conforme mostrado na Figura 3.5.

Figura 3.5: Autômato do serviço *end_session*.

O serviço *list_status* é utilizado para requisitar a lista de conteúdo disponível de outro *peer*. Esse serviço é utilizado apenas por *peers* consumidores. Como resultado, a aplicação obtém uma lista com os identificadores dos conteúdos do outro *peer*.

Para isso, a aplicação P2P envia uma mensagem *ls_req* para a camada CAL que, por sua vez, envia a mensagem *send_msg_request* com o *ls_req* dentro do seu campo de dados e, então, espera pela confirmação que pode ser positiva ou negativa. Se for negativa, a camada volta para o estado *CONNECTED* e envia a mensagem *ls_conf-* para a aplicação. Caso a mensagem recebida seja uma confirmação positiva, a camada CAL envia uma mensagem *send_msg_resp+* para confirmar o recebimento da mensagem sem falhas ou *send_msg_resp-* para indicar alguma falha. Essa mensagem de confirmação também é enviada dentro do campo de dados da primitiva *send_msg_req*.

Nesse momento, a camada CAL torna-se pronta para receber a lista dos identificadores. Em alguns casos, o *peer* possui uma quantidade de conteúdo tão grande que a lista deve ser dividida em pedaços menores para ser enviada, conforme mostrado na Figura 3.6. Nesse caso, cada segmento é enviado com a *flag* *MORE_BIT* com o valor igual a 1 até que todos os segmentos sejam enviados. O autômato desse serviço também considera um tempo limite para recebimento da mensagem *ls_conf*, para o envio e também para o recebimento de um segmento.

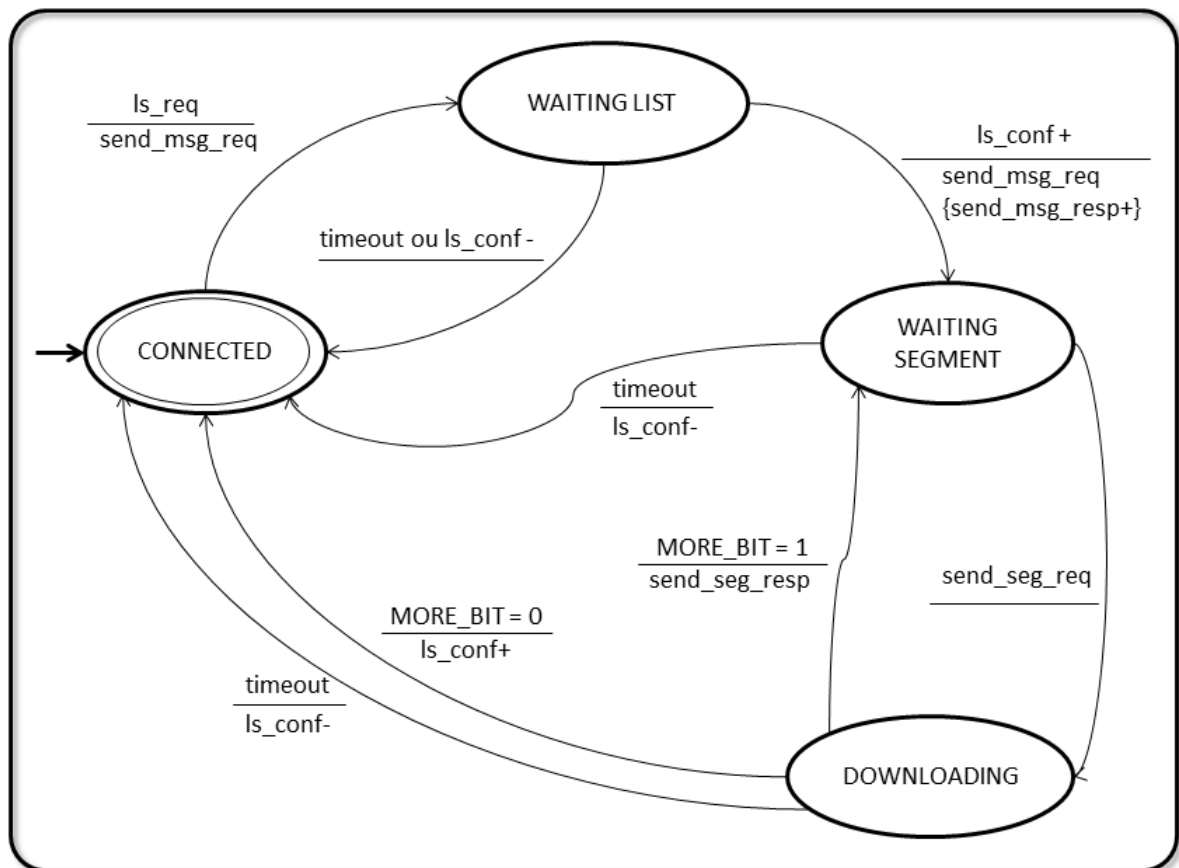
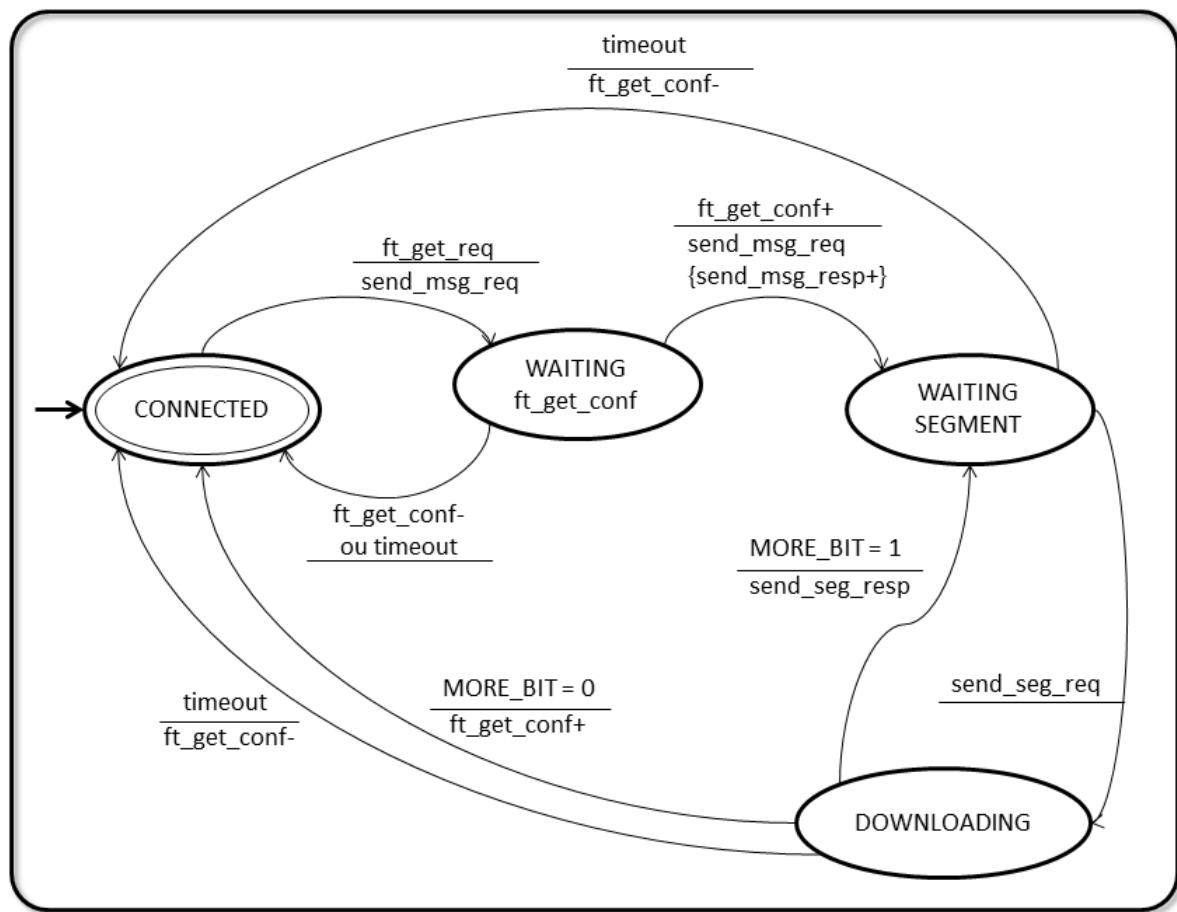


Figura 3.6: Autômato do serviço *list_status*.

Para um melhor entendimento, a Figura 3.7 mostra um diagrama de ordem temporal para um cenário que utiliza o serviço *list_status*. É importante ressaltar que a camada CAL utiliza o campo de dados da primitiva *send_msg* do JXTA para enviar as mensagem

Figura 3.8: Autômato para o serviço *ft_get*.

o mesmo faz uma chamada desse serviço para armazenar o conteúdo capturado.

Assim, para que o conteúdo seja enviado, a aplicação P2P já deve estar conectada a outro *peer* para enviar a mensagem *ft_put_req*. Ao receber essa mensagem, a camada envia a solicitação para o outro *peer* através do serviço *send_msg* e, então, espera a confirmação do *peer* consumidor. Caso receba uma confirmação negativa, a camada sinaliza a aplicação que não foi possível enviar o conteúdo e volta para o estado CONNECTED. Essa situação pode acontecer, por exemplo, no caso do *peer* consumidor não possuir mais espaço disponível para armazenamento. Com isso, a decisão de tentar enviar o conteúdo novamente para o mesmo *peer* ou então tentar uma outra conexão é uma responsabilidade da aplicação. Caso a confirmação seja positiva, a camada se torna pronta para enviar os segmentos do conteúdo. No momento em que todos os segmentos forem enviados (isto é, $\text{MORE_BIT} = 0$), a camada CAL volta para o estado CONNECTED e envia a mensagem *ft_put_conf+* para a aplicação, indicando que o *upload* do conteúdo foi efetuado corretamente. Esse comportamento pode ser melhor visualizado no autômato apresentado na Figura 3.9.

O serviço *send_seg* é responsável por enviar cada parte do conteúdo que foi dividido em segmentos. Esse serviço está disponível tanto para produtores quanto para consumidores. Principalmente no cenário de compartilhamento de conteúdo multimídia, normalmente

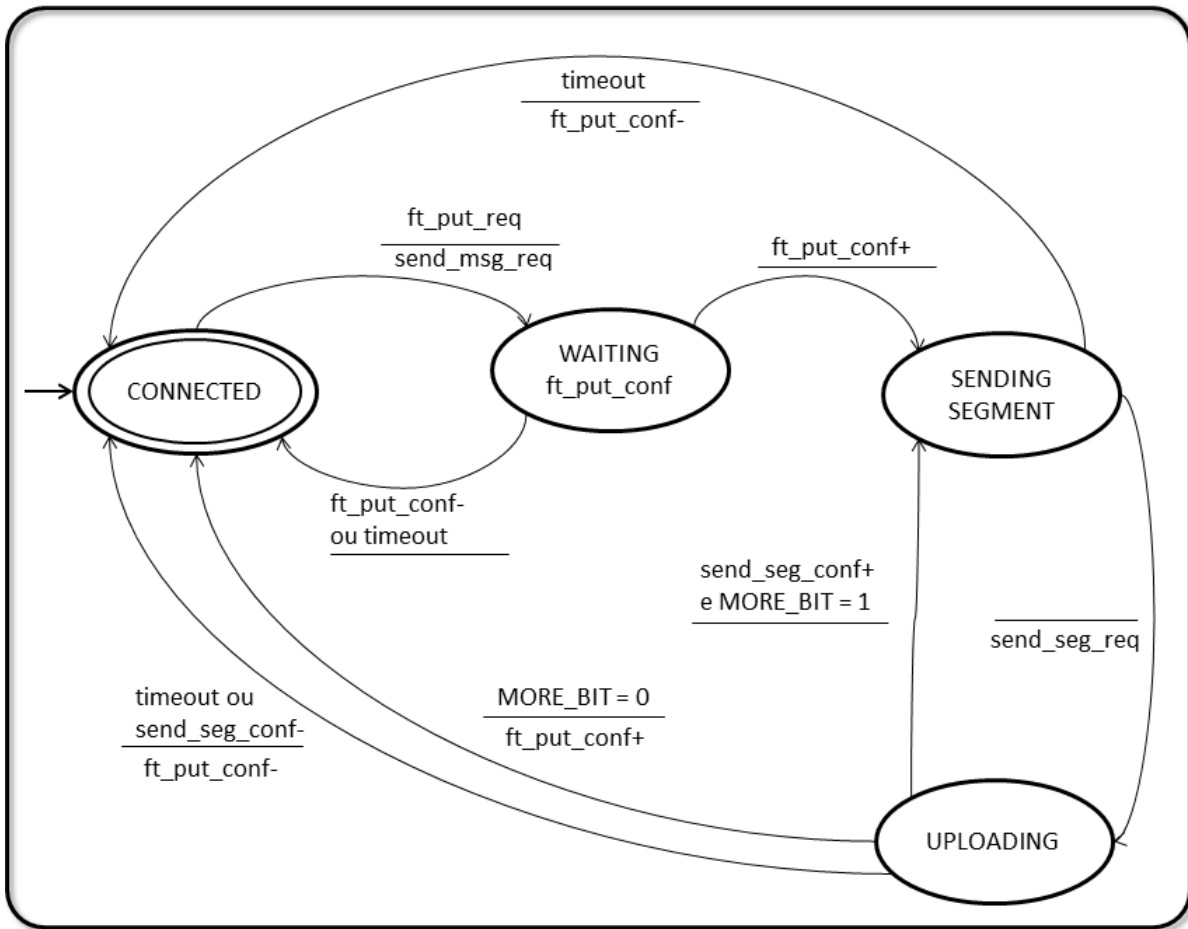


Figura 3.9: Autômato para o serviço *ft_put*.

os conteúdos são muito grandes para serem transferidos em uma única vez. Assim, eles são quebrados e enviados em partes menores. O comportamento desse serviço pode ser visualizado nos autômatos dos serviços *list_status*, *ft_get* e *ft_put* apresentados anteriormente.

Finalmente, o último serviço é chamado de *reject* e também está disponível tanto para produtores quanto para consumidores. Esse serviço é utilizado para desconsiderar primitivas com erros ou, ainda, primitivas cujo serviço não está disponível para o *peer* em questão, ou seja, mensagens que são reconhecidas somente por *peers* que exercem uma determinada função. Nesses casos, esse serviço deve ser usado para rejeitar as primitivas recebidas erroneamente. Eventualmente, esses erros podem causar efeitos colaterais na camada CAL pelo fato da possibilidade de existir rajadas de erro que a camada de enlace não seja capaz de lidar em consequência de falsos positivos.

O comportamento do serviço *reject* é bem simples. Em todos os estados dos serviços apresentados anteriormente, a camada CAL pode invocar esse serviço para sinalizar o erro (caso receba uma primitiva errada) e então continuar no mesmo estado esperando por outra mensagem. Dessa forma, a consistência da comunicação entre a aplicação P2P e a camada CAL é assegurada.

3.3 Considerações Finais

Neste capítulo, a estrutura detalhada do modelo de comunicação utilizado foi apresentado. Ela foi projetada para transferir e replicar todo o conteúdo multimídia capturado por diferentes dispositivos instalados em ambientes instrumentados.

A camada CAL cria abstrações para descobrir *peers* que oferecem serviços de armazenamento com propriedades de segurança e confiabilidade. Assim, os dispositivos que produzem conteúdo multimídia não precisam conhecer previamente os *peers* existentes na rede para transferir o conteúdo capturado, como o endereço IP ou um identificador único (ID).

Os elementos essenciais do protocolo P2P utilizado pela camada CAL também foram apresentados. Esse protocolo foi projetado de maneira genérica, para que o mesmo também pudesse ser utilizado em outros domínios, que não o educacional. Além disso, o protocolo permite que as próprias aplicações P2P gerenciem suas políticas de transferência/replicação.

Capítulo 4

Uma Arquitetura para Acesso Personalizado de Conteúdo Multimídia

Outro desafio já conhecido associado às aplicações de C&A está relacionado com a gestão da enorme quantidade de conteúdo gerado, além de seu crescimento ser contínuo.

Considere, por exemplo, o domínio educacional com a captura de diversas atividades em salas de aula equipadas com uma lousa eletrônica, microfones e câmeras de vídeo. Em tal cenário, diversos artefatos multimídia podem ser gerados a partir da gravação de uma única sessão de captura – *slides* da lousa eletrônica, áudio dos microfones e vídeo das câmeras. Com todo esse conteúdo disponível, os alunos se interessam em acessar partes relevantes do mesmo, procurando artefatos de mídia específicos de acordo com suas necessidades e gostos específicos em mente.

Assim, as informações armazenadas em sistemas automatizados de C&A são, portanto, inúteis se os usuários não conseguirem encontrar facilmente o que procuram. Funcionalidades como recomendação de conteúdo, classificação, personalização e suporte a preferências pessoais são altamente desejáveis pelos usuários de tais sistemas.

Um exemplo prático dessas demandas pode ser representado por um estudante que deseja encontrar materiais relacionados às avaliações de uma disciplina. Com base em seu contexto durante o acesso, por exemplo, local do acesso, tempo disponível para estudos e recursos do dispositivo eletrônico utilizado, o sistema poderia oferecer-lhe uma experiência de acesso personalizada, fornecendo uma interface que aponta diretamente para o conteúdo da próxima avaliação agendada.

Além disso, a personalização neste caso poderia ir além, adaptando os conteúdos já filtrados de acordo com restrições mais específicas, como excluir o conteúdo de áudio porque os fones de ouvido do usuário não estão disponíveis naquele dispositivo ou, ainda, mostrando apenas os *slides* capturados sem o vídeo porque a velocidade da conexão é

baixa.

Dessa forma, o contexto de acesso dos usuários juntamente com suas preferências pessoais e as restrições de apresentação do dispositivo devem ser exploradas a fim de proporcionar uma experiência de acesso personalizada e mais interativa. Assim, esse capítulo apresenta uma proposta de Arquitetura de Acesso Contextual, em inglês *Contextual Access Architecture (CAA)*, que permite recomendação de conteúdo, *ranking* e personalização de apresentações multimídia interativas capturadas em salas de aula instrumentadas.

Este capítulo é organizado da seguinte maneira: a Seção 4.1 mapeia as variáveis do modelo de contexto de acesso utilizado de acordo com a literatura; a Seção 4.2 mostra a abordagem utilizada para a utilização de preferências pessoais contextuais; a Seção 4.3 detalha o tratamento de restrições de apresentação; a Seção 4.4 apresenta a arquitetura proposta em detalhes; e, por fim, a Seção 4.5 traz as considerações finais do capítulo.

4.1 Mapeamento das Dimensões de Contexto

Diversas definições de *contexto* são encontradas na literatura. Dentre elas, encontra-se a definição dada pelos primeiros a utilizar o termo “ciência de contexto”, ou *context-awareness* em inglês, Schilit e Theimer (1994) em 1994. Para eles, *contexto* é definido por algumas variáveis como localização, pessoas e objetos próximos, e ainda suas mudanças ao longo do tempo. Outros autores definem *contexto* de uma forma similar, porém removendo ou adicionando outras variáveis como hora do dia, temperatura, estação do ano, entre outras.

Abowd *et al.* (1999) definem *contexto* como qualquer informação que descreva uma entidade. Para eles, uma entidade pode ser uma pessoa, um lugar ou um objeto. Os mesmos autores definem “*context-awareness*” como o uso do contexto para fornecer informações relevantes e/ou serviços para os usuários.

Em computação móvel e ubíqua, o contexto do usuário se altera muito rapidamente. Muitos aplicativos utilizam variáveis de contexto para personalizar a apresentação do conteúdo aos usuários de acordo com sua situação atual. Desta forma, o contexto é essencial para proporcionar uma melhor experiência do usuário ao recuperar o conteúdo previamente capturado.

De acordo com Truong *et al.* (2001), há um conjunto mínimo de perguntas que devem ser consideradas ao projetar aplicações de C&A cientes de contexto:

- *Quem* são os usuários?
- *O que* é capturado e acessado?
- *Quando* a captura e o acesso ocorrem?
- *Onde* a captura e o acesso ocorrem?

- *Como* realizar a captura e o acesso?

Baseado no mapeamento original de Truong *et al.* (2001) e Abowd *et al.* (1999), a arquitetura CAA considera 7 dimensões para a modelagem do contexto dos usuários em aplicações de C&A para ambientes educacionais. Essas dimensões podem ser facilmente estendidas de acordo com as necessidades de cada aplicação. As 7 dimensões de contexto consideradas na fase de acesso são:

- O *tipo de dispositivo*, por exemplo, “Desktop”, “Tablet” ou “Smartphone”;
- A *velocidade de conexão* do dispositivo, medida em Kbps;
- A *resolução de tela* do dispositivo, por exemplo, “até 240x360”, “até 800x600” ou “maior que 800x600”;
- A *data e a hora* do acesso;
- O *tempo disponível* do usuário para visualização do conteúdo capturado, por exemplo, “até 15 minutos”, “até 30 minutos”, “até 60 minutos” ou “mais de 60 minutos”;
- O *local* onde o usuário está realizando o acesso, por exemplo, “Universidade”, “Casa” ou “Trabalho”; e,
- O *motivo* pelo qual o usuário está acessando o sistema, por exemplo, “Estudos normais”, “Avaliação”, “Revisão” ou “Aula perdida”.

A Figura 4.1 mostra cada uma das dimensões de contexto apresentadas com seus respectivos valores de exemplo. É importante notar que as dimensões referentes às perguntas “*Quem* são os usuários?” e “*O que* é capturado e acessado?” são implícitas, visto que os usuários são os alunos e os objetos capturados e acessados são as aulas.

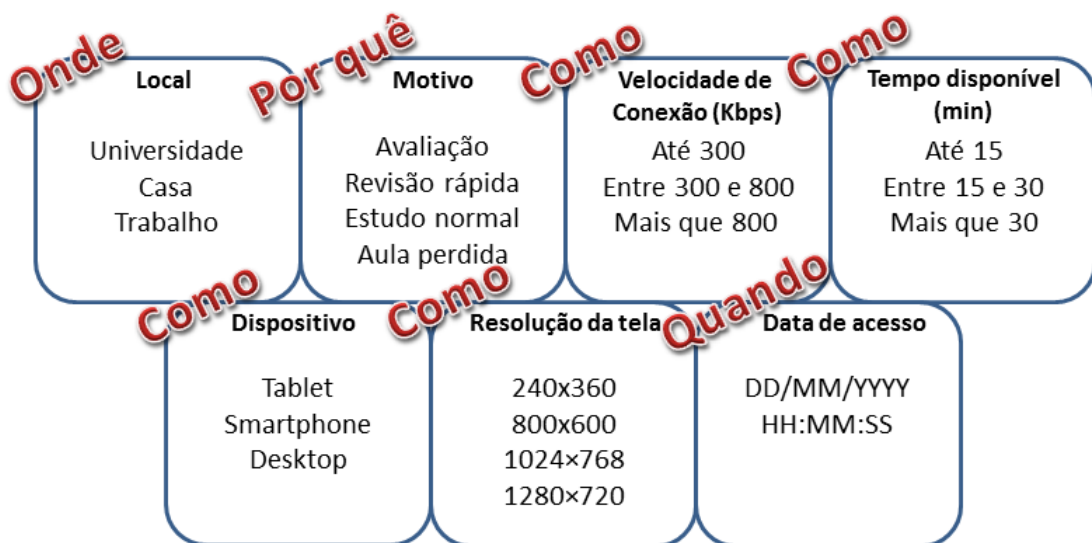


Figura 4.1: Contexto de acesso do usuário.

Dessa forma, o contexto de acesso é representado pela tupla c que contém um identificador único, o usuário em questão e os atributos de contexto:

$$c = (id, usuario, dispositivo, banda, resolucao, data, disponibilidade, local, motivo)$$

4.2 Preferências Contextuais de Usuários

As preferências do usuário são um conjunto de preferências pessoais utilizadas para recomendar o conteúdo capturado que melhor se adequa às suas necessidades.

Um estudante que prefere visualizar as aulas mais vistas que estão relacionadas à uma avaliação seria um bom exemplo de tal demanda. Com base em seu contexto de acesso, o sistema poderia proporcionar-lhe uma experiência de acesso personalizada, apresentando o conteúdo diretamente relacionado com o assunto da próxima avaliação programada, de acordo com as estatísticas de acesso dos outros alunos. Além disso, se a data de acesso é próxima à data da avaliação e o tempo disponível para estudos é pequeno, apenas o *slides* das aulas mais importantes poderiam ser apresentados ao invés de todo o conteúdo (*slides*, áudio e vídeo) de todas as aulas.

Da mesma forma, um aluno com deficiência visual poderia registrar uma preferência pessoal no sistema afirmando que ele sempre prefere a reprodução de áudio a apresentação de *slides* das aulas. Tais preferências são válidas para todas as disciplinas matriculadas.

O principal problema enfrentado em trabalhos que lidam com preferências pessoais é identificar quais as preferências que são mais relevantes para realizar uma consulta contextual (uma consulta padrão aprimorada com o contexto do usuário) e apresentar um algoritmo para localizá-las. Baseado nisso, a abordagem proposta pela CAA visa localizar as melhores preferências em um repositório através de uma consulta SQL que considera o contexto de acesso de cada usuário.

Por um lado, os usuários podem especificar as suas preferências pessoais de acordo com o contexto de acesso. Por outro lado, os valores das variáveis de contexto são inerentes a uma situação fora do banco de dados e, em geral, não podem ser afetados pelas preferências do usuário.

Assim, a arquitetura dispõe de um repositório para armazenamento das preferências que podem ser selecionadas por diferentes abordagens e, posteriormente, utilizadas para filtrar e classificar o conteúdo a ser disponibilizado.

Pode-se, ainda, utilizar algoritmos para encontrar a melhor maneira de responder a uma consulta do usuário, mesmo quando ele não informar explicitamente suas preferências, bastando popular e atualizar o “Repositório de Preferências” ao longo do tempo.

Em síntese, várias abordagens que tratam de preferências de usuários podem ser aplicadas como, por exemplo, a abordagem de consultas personalizadas proposta por Koutrika e Ioannidis (2010) ou ainda as abordagens apresentadas por Stefanidis *et al.* (2011), Liang

et al. (2007) e de Amo e Ribeiro (2009).

O objetivo da arquitetura CAA não é propor um algoritmo ou método específico, mas sim apoiá-los como mecanismos alternativos para coletar informações do usuário e identificar suas preferências pessoais para que o conteúdo seja filtrado de acordo com consultas personalizadas.

4.3 Restrições de Apresentação

As restrições de apresentação são utilizadas em complemento à abordagem de utilização de contexto de acesso e de preferências contextuais de usuários. Enquanto as preferências dos usuários se referem aos requisitos de informações dos usuários, as restrições de apresentação se referem ao domínio do sistema e do ambiente.

Um exemplo é o caso de um usuário que realiza o acesso ao conteúdo através de um dispositivo cuja velocidade de conexão seja baixa ou, ainda, de um estudante que deseja fazer uma revisão rápida antes de uma próxima aula. Em ambos os casos, exibir o fluxo de vídeo não é recomendado. Assim, o conteúdo capturado deve ser apresentado de forma personalizada com apenas os *slides* da aula e sem o vídeo.

Outra situação seria, por exemplo, um aluno usando um dispositivo móvel com uma baixa resolução de tela. Talvez esse aluno não veja muito bem os detalhes dos *slides* das aulas e, nesta situação, o sistema poderia recomendar apenas o áudio da aula.

Estes exemplos mostram apenas algumas das possíveis restrições de apresentação que podem ser impostas levando em consideração o contexto de acesso do usuário e suas preferências pessoais.

Tais restrições são impostas ao usuário apenas sugerindo possibilidades de personalização para a apresentação do conteúdo multimídia recuperado. Isto é feito através da classificação (*ranking*) de possíveis documentos para serem exibidos e o usuário é livre para escolher aquela que melhor se adapta às suas necessidades a fim de visualizar a aula previamente capturada.

As restrições de apresentação são armazenadas em documentos XML e contêm um nó chamado de *antecedente* e um outro nó chamado de *consequente*. O *antecedente* é um conjunto de condições que satisfazem completamente a restrição em questão, considerando um peso que é dado a cada uma dessas condições. Esse peso é personalizável de acordo com cada restrição de apresentação, ou seja, ele não possui um valor fixo para todas as restrições. O *consequente* contém o nome da melhor folha de estilo a ser aplicada na apresentação do conteúdo capturado, caso as condições do *antecedente* sejam satisfeitas.

O Fragmento 4.1 apresenta um exemplo de uma restrição de apresentação cuja semântica indica que a melhor forma de apresentação do conteúdo é aquela em que somente os *slides* são exibidos, caso: (1) o usuário acesse o sistema com a velocidade de conexão menor que 300Kbps, (2) possua menos que 30 minutos disponíveis para o estudo, (3) a

resolução da tela seja maior ou igual a 800x600 *pixels* e (4) o motivo do acesso seja revisão rápida. Além disso, a dimensão de contexto “*resolução da tela*” recebeu um peso maior com relação às outras dimensões para o cálculo do *ranking*.

Fragmento 4.1: Exemplo de uma restrição de apresentação.

```
<rule>
  <antecedent>
    <condition att="connection_speed" op="!" val="300" weight="0.2"></condition>
    <condition att="available_time" op="!" val="30" weight="0.2"></condition>
    <condition att="screen_resolution" op=">" val="800x600" weight="0.4"></condition>
    <condition att="motive" op="=" val="quick_review" weight="0.2"></condition>
  </antecedent>
  <consequent>
    <result name="Somente_slides" val="xhtml_slides.xml"></result>
  </consequent>
</rule>
```

4.4 Visão Geral da Arquitetura

Os três conceitos apresentados nas seções anteriores foram unificados na abstração da arquitetura CAA. Essa arquitetura foi projetada para ser utilizada em aplicações de C&A de diversos domínios, com suporte a recomendação e a personalização de conteúdo considerando contexto de acesso, preferências contextuais dos usuários e restrições de apresentação.

Todos os módulos dessa arquitetura são fracamente acoplados, o que significa que a arquitetura foi projetada para permitir diferentes implementações de algoritmos e/ou métodos para recomendação e personalização de conteúdo. A Figura 4.2 mostra os módulos que compõem a arquitetura.

O primeiro módulo, chamado “Módulo de Coleta e Inferência de Contexto”, é responsável por coletar as informações relacionadas ao contexto do usuário ou executar inferências básicas sobre ele. Dependendo de quão sofisticada for a capacidade de detecção da aplicação, o contexto pode ser coletado automaticamente, informado explicitamente pelo usuário, inferido implicitamente pelo sistema ou uma combinação dessas abordagens (forma híbrida). O importante é que, uma vez que o contexto esteja definido, ele se torna a unidade primária de informação passada através da CAA.

Com base nas informações de contexto, as preferências do usuário e as restrições de apresentação são selecionadas em seguida. Dois repositórios correspondentes armazenam as preferências e as restrições separadamente. Mais uma vez, independentemente da abordagem utilizada para o tratamento de preferências, a arquitetura suporta qualquer utilização. Quanto às restrições de apresentação, elas podem ser definidas manualmente pelos usuários mas o sistema também pode fornecer algumas restrições de apresentação padrão, que são válidas para todos os usuários.

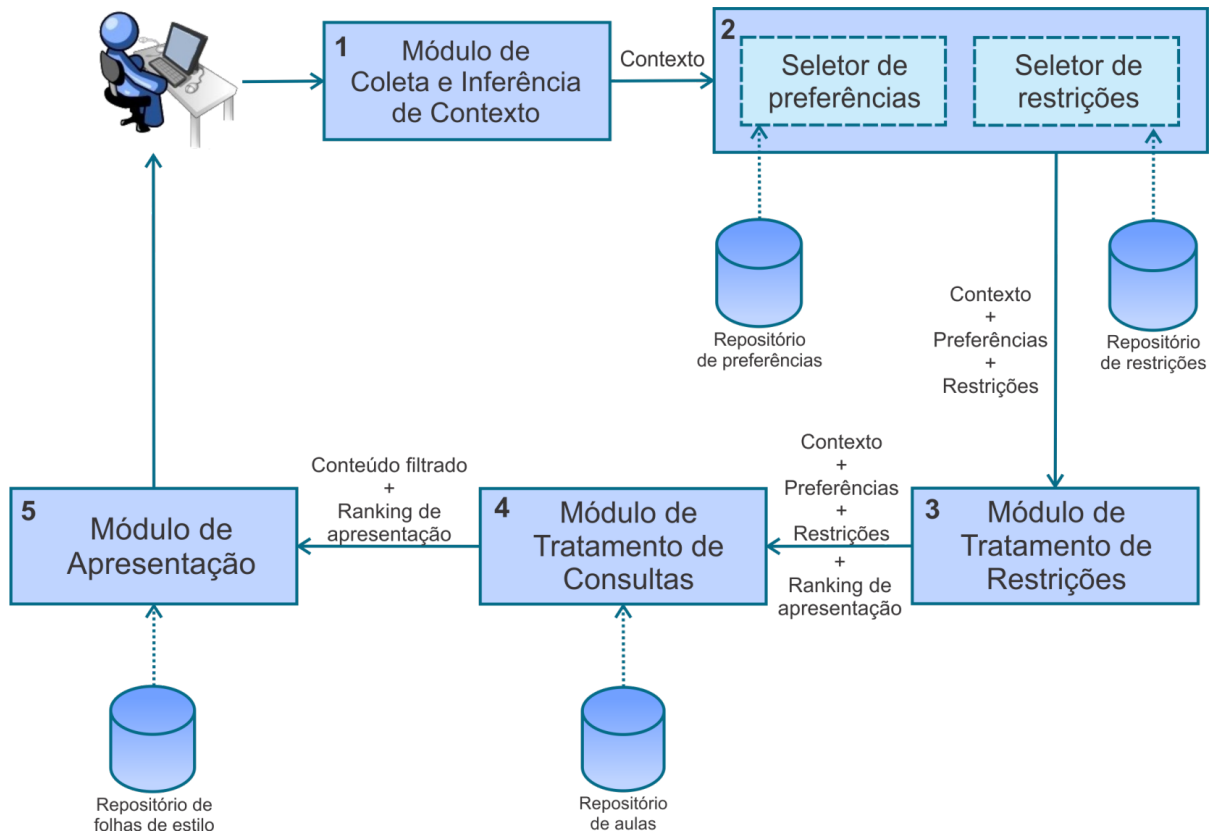


Figura 4.2: Visão Geral da Arquitetura de Acesso Contextual (CAA).

Uma vez que o sistema esteja ciente do contexto do usuário, de suas preferências e de todas as restrições de apresentação relevantes, o próximo módulo, nomeado “Módulo de Tratamento de Restrições”, processa essas restrições com base no contexto do usuário e gera um *ranking* para apresentação personalizada do conteúdo multimídia. Aqui, novamente, outros algoritmos, métodos ou modelos para classificação das diversas formas de apresentação podem ser aplicados a essas restrições como, por exemplo, o método de *ranking* baseado em análise de *logs* e o modelo proposto por Kim *et al.* (2012) baseado em *folksonomy*.

Ainda com base no contexto do usuário, suas preferências e restrições, o “Módulo de Tratamento de Consultas” é responsável por reescrever a consulta que será submetida ao banco de dados para selecionar os melhores resultados de conteúdo para o usuário em questão. Esse módulo procura pelas aulas no “Repositório de Aulas” e retorna um conjunto já filtrado de conteúdo multimídia.

Por fim, o “Módulo de Apresentação” se encarrega de apresentar corretamente o conjunto resultante do conteúdo capturado para o usuário. Esse módulo recupera os *layouts* de apresentação no “Repositório de Folhas de Estilo” e cria um *ranking* dos mesmos, classificando as formas de apresentação com mais chances de atender às necessidades do usuário. No entanto, as outras opções de visualização da aula estarão disponíveis para que o usuário sinta-se livre para escolher qualquer uma delas.

4.5 Considerações Finais

A Arquitetura de Acesso Contextual (CAA) foi apresentada como uma proposta para facilitar o desenvolvimento de aplicações de captura multimídia automatizadas nas quais as preferências dos usuários e seu contexto de acesso são relevantes para produzir visões personalizadas de artefatos de mídia existentes. Ela suporta a recomendação de conteúdo e a classificação (*ranking*) em tempo de projeto, sendo uma ferramenta útil para ambientes educacionais ubíquos.

Além disso, a CAA é composta por componentes fracamente acoplados que podem ser substituídos de acordo com as necessidades de cada aplicação e ainda pode ser estendida através da possibilidade de cada módulo ser implementado de diferentes maneiras (por exemplo, o algoritmo de *ranking*).

O próximo capítulo apresentará o funcionamento detalhado da CAA juntamente com seus principais aspectos de projeto e implementação, que serão mostrados através da sua integração em uma aplicação de C&A projetada para o domínio educacional.

Capítulo 5

Classroom eXperience: Estudo de Caso

Como estudo de caso, as abordagens propostas para acesso contextual e comunicação foram implementadas em uma aplicação real chamada de *Classroom eXperience (CX)*¹, que é uma plataforma de captura multimídia concebida para gravar automaticamente as aulas ministradas e disponibilizá-las posteriormente aos alunos.

Com o CX é possível capturar, armazenar e acessar o conteúdo multimídia capturado em salas de aula instrumentadas com lousas eletrônicas, microfones, câmeras de vídeo e projetores. Os fluxos de mídia de cada dispositivo são capturados e sincronizados, gerando documentos hipermídia em diferentes formatos de apresentação.

Construído sobre a plataforma iClass (Pimentel *et al.*, 2007), o CX complementa seu alicerce implementando a camada CAL, como meio de comunicação, e também a arquitetura CAA, para personalização e recomendação de conteúdo. Tal abordagem permite a personalização da apresentação do conteúdo capturado de acordo com as preferências e o contexto de acesso de cada estudante.

Sem alterar a dinâmica convencional das aulas, o CX utiliza recursos de computação ubíqua para auxiliar tanto instrutores no processo de captura de aulas quanto estudantes no processo de acesso ao conteúdo ministrado.

O CX conta com componentes de software especializados para gravar as atividades educacionais, sincronizar os diferentes fluxos de mídia gerados e disponibilizar o conteúdo através de uma interface amigável. Todos os componentes foram desenvolvidos utilizando a linguagem Java com suporte a bibliotecas Web.

Além disso, as informações da aplicação bem como o conteúdo gerado são armazenados no banco de dados PostgreSQL² e, atualmente, possui 23 tabelas. O Diagrama Entidade-Relacionamento (DER) da base de dados do CX é mostrado no Apêndice A.

¹<http://cx.facom.ufu.br>

²<http://www.postgresql.org>

As subseções seguintes apresentam os aspectos de concepção, implementação e funcionamento do CX de acordo com as fases propostas por Abowd *et al.* (1996) para a estruturação do problema de C&A. As fases de pós-produção e acesso serão mais enfatizadas, visto que, na primeira, emprega-se o modelo de comunicação desenvolvido para replicação de conteúdo e, na segunda, faz-se uso da arquitetura contextual proposta para recomendação e personalização do conteúdo capturado.

5.1 Pré-produção

No CX, a fase de pré-produção é responsável por definir o conteúdo que será capturado em sala de aula. Nessa fase, o instrutor registra todas as informações relevantes da aula através de uma interface Web. As informações da aula são organizadas em três categorias: dados básicos, dados adicionais (meta-informação) e o próprio conteúdo que será apresentado aos alunos.


A primeira categoria está relacionada com as informações básicas da aula como título, palavras-chave, resumo, duração, etc. Essas informações são armazenadas em um banco de dados relacional e contém *links* para as outras informações.

A segunda categoria está relacionada com as informações adicionais da aula. Essas informações são classificadas e organizadas através de extensões baseadas no padrão *IEEE Learning Object Metadata (LOM)* (IEEE, 2002). O Anexo A apresenta a estrutura do objeto LOM referente a uma aula específica.

A terceira e última categoria é o conteúdo em si que será apresentado em sala de aula. Esse conteúdo é um arquivo de *slides* que pode estar nos formatos pdf, ppt ou pptx. Um componente de software especializado se encarrega de transformar esse arquivo em um *template* reconhecido pelo sistema de captura.

Após o processamento das informações dessa fase, o conteúdo já está pronto para ser apresentado em sala de aula através do processo de gravação ao vivo, que é apresentado em seguida.

A Figura 5.1 mostra a página Web onde o instrutor registra as informações de uma nova aula que será gravada posteriormente em sala de aula.

Luiz Carlos Andrada

Início > GBC037 - Interação Humano-Computador > Nova Aula

Informe os dados da nova aula:

Título*:

Palavras-chave*:

Arquivo de Slides: Nenhum arquivo selecionado
(.pdf, .ppt, .zip)

Resumo:

Duração: min.

Língua:

Nível de dificuldade:

Importância:

☐ LOM (Learning Object Metadata)

Tipo interatividade

Recurso de aprendizado

Nível de interatividade

Regra

Contexto

Idade

Tempo de aprendizado

Formato

Tamanho

Local

Custo

Direitos

Descrição

Versão

Status

Contribuição

*Preencha os campos obrigatórios

© Copyright 2012-2013 UbiMedia · Todos os direitos reservados



 Português  Sobre

Figura 5.1: Criação de uma nova aula no *Classroom eXperience*.

5.2 Gravação ao vivo

Um dos grandes desafios ao projetar aplicações de C&A é fazer com que a fase de gravação ao vivo aconteça de forma transparente para os usuários envolvidos. Isso significa que o processo de captura deve ser o mais discreto possível, com alterações mínimas na rotina do instrutor.

Essa fase acontece quando a aula é apresentada aos alunos em sala de aula. Para dar início ao processo de captura, o instrutor acessa o sistema através de uma interface Web e seleciona a aula desejada. Nesse momento, o seu conteúdo é automaticamente carregado na lousa eletrônica. Desse ponto em diante, apesar do instrutor estar usando uma caneta eletrônica ao invés de giz, a aula acontece sem nenhuma interferência do sistema.

Em segundo plano, um aplicativo de captura faz o *download* de todo o conteúdo preparado e intermedeia a comunicação entre os dispositivos de captura disponíveis no ambiente. Esse aplicativo de captura é, na verdade, um módulo P2P (Araújo *et al.*, 2012) composto por três componentes de software responsáveis por capturar cada uma das diferentes mídias: o componente de captura de áudio, o componente de captura de vídeo e o componente de captura das anotações feitas sobre a lousa eletrônica.

A Figura 5.2 mostra uma sala de aula instrumentada com lousa eletrônica, dois projetores multimídia, microfone e câmera de vídeo, que utilizam toda a infraestrutura do CX. O instrutor pode interagir com a lousa eletrônica como se ele estivesse utilizando uma lousa tradicional ou um quadro de giz. Todos os traços escritos na superfície da lousa eletrônica são projetados em uma segunda tela para facilitar a visualização do conteúdo pelos alunos.



Figura 5.2: Aula sendo capturada com a infraestrutura do *Classroom eXperience*.

5.3 Pós-produção

A fase de pós-produção se inicia imediatamente após o término da captura ao vivo. Nessa fase, a aplicação de captura reúne todos os fluxos de mídias capturados e realiza a

sincronização dos mesmos. Todas as informações capturadas são sincronizadas através de documentos XML (*eXtensible Markup Language*) contendo as marcações das interações ocorridas entre o instrutor e o sistema de captura.

Além disso, alguns documentos são gerados para que o conteúdo seja apresentado aos usuários. Esses documentos são gerados em dois formatos padrão: (1) HTML, para visualização Web e (2) NCL, usado pela Plataforma Brasileira de TV Digital Interativa (Silva *et al.*, 2004).

A fase de pós-produção é composta por três componentes especializados de software:

- **Componente de sincronização:** esse componente é responsável por reunir os fluxos de mídia capturados em um único documento XML contendo informações sobre o contexto da aplicação, sobre os dispositivos de captura e as interações ocorridas entre o sistema de captura e os dispositivos;
- **Componente de transformação:** esse componente é responsável por gerar os diversos formatos para apresentação do conteúdo capturado. Para isso, ele utiliza o XML criado pelo componente de sincronização para transformá-lo em documentos HTML e NCL utilizando folhas de estilo XSLT³ (*eXtensible Stylesheet Language for Transformation*); e, por último,
- **Componente de compactação:** é responsável por reunir todos os documentos gerados pelos componentes anteriores e compactá-los em um formato padrão, reconhecido pela etapa de acesso.

Após ser compactado pelo componente responsável, o conteúdo deve ser armazenado para sua posterior disponibilização. Nesse momento, a aplicação utiliza a camada CAL para realizar a transferência. É importante ressaltar que a aplicação não sabe onde o conteúdo será armazenado. Ela apenas faz uma chamada ao serviço de armazenamento da camada CAL que se encarrega de distribuí-lo aos *peers* ativos na rede.

5.4 Acesso

A fase de acesso consiste na apresentação do conteúdo capturado aos alunos. Todo o conteúdo multimídia que foi capturado durante a aula deve ser apresentado de uma forma clara e organizada.

De acordo com Kientz (2012), muitas vezes o processo de acesso ao conteúdo nas aplicações de C&A é tratado em segundo plano e o esforço investido em seu desenvolvimento não é suficiente. Se a fase de acesso for desenvolvida com foco nos usuários, eles serão mais incentivados a visualizar o conteúdo previamente capturado.

Assim, bons recursos para recuperação, recomendação e personalização de conteúdo são fundamentais para o sucesso de qualquer sistema de C&A.

³www.w3.org/TR/xslt

5.4.1 Aspectos de Usabilidade da Interface de Acesso e Apresentação

Para que as propostas apresentadas pudessem ser disponibilizadas aos usuários finais durante a visualização do conteúdo capturado, realizou-se um estudo de usabilidade para que a interface de acesso e apresentação do CX fosse projetada e implementada considerando tais demandas, buscando meios de incentivar ainda mais os alunos a acessar o conteúdo para estudo. A nova interface se apoia nos pilares de um bom projeto de interação entre humanos e computadores (IHC): princípios, padrões e *guidelines* (Dix *et al.*, 2004), em especial os princípios representados pelas 10 heurísticas de Nielsen (Nielsen, 1993), a saber:

1. Visibilidade do status do sistema;
2. Compatibilidade do sistema com o mundo real;
3. Controle do usuário e liberdade;
4. Consistência e padrões;
5. Prevenção de erros;
6. Reconhecimento ao invés de relembração;
7. Flexibilidade e eficiência de uso;
8. Estética e design minimalista;
9. Ajuda aos usuários no reconhecimento, diagnóstico e correção de erros; e
10. Ajuda e documentação.

A priori, um protótipo de baixo nível foi criado principalmente com base nas heurísticas “visibilidade do status do sistema”, “reconhecimento ao invés de relembração”, “controle do usuário e liberdade” e “consistência e padrões”, características que podem ser observadas nos esboços desenvolvidos.

A Figura 5.3 apresenta um esboço da tela principal de uma disciplina, desenvolvido inicialmente como protótipo de interface do *Classroom eXperience*. Na figura, é possível perceber a forma com que as aulas cadastradas na disciplina selecionada serão exibidas. Há ainda a possibilidade de visualizar cada aula no formato HTML ou realizar o *download* do arquivo NCL e a parte lateral direita do esboço possui uma área onde são exibidas as atualizações do sistema referentes àquela disciplina como, por exemplo, informações sobre novas aulas ministradas.

A Figura 5.4 ilustra outro esboço de tela no qual pode-se observar a disposição do conteúdo multimídia assim que o usuário seleciona uma aula no formato HTML para ser visualizada. Nessa figura, observa-se uma barra inferior na qual uma miniatura dos *slides* da aula são exibidos de forma horizontal e contínua, com setas para melhorar a navegação,

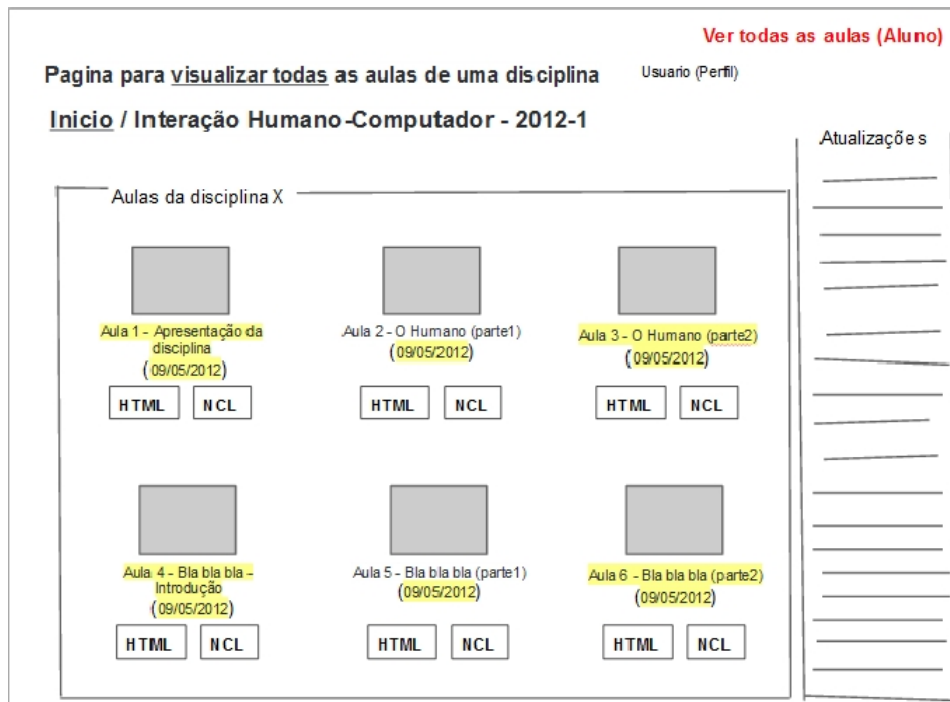


Figura 5.3: Protótipo da tela de listagem das aulas de uma disciplina.

permitindo que o usuário tenha ciência sobre a quantidade de *slides* que a aula possui. A parte lateral esquerda é constituída pelo vídeo que foi registrado durante a aula e uma área onde comentários sobre a aula selecionada podem ser escritos (funcionalidade descrita nos trabalhos futuros). Além disso, setas direcionais de controle foram adicionadas na área de exibição principal de slides, facilitando a navegação entre *slides* sequenciais.

Após o desenvolvimento do protótipo de baixa fidelidade, um protótipo de alta fidelidade foi desenvolvido utilizando a ferramenta *Adobe Flash* no intuito de utilizá-la durante as avaliações de usabilidade feitas com especialistas e usuários.

A avaliação heurística, proposta por Nielsen (1993), foi a técnica de avaliação de usabilidade escolhida, realizada por profissionais da área de interação e usabilidade. Este método utiliza um conjunto de princípios de usabilidade, chamadas de heurísticas, para avaliar os elementos da interface de um determinado sistema. De acordo com Nielsen, cinco avaliadores são suficientes para encontrar cerca de 75% dos problemas de usabilidade existentes e, por isso, cinco profissionais foram convidados a fazer a avaliação do CX.

As falhas de usabilidade mais significativas foram identificadas na tela principal do sistema e na tela de visualização dos *slides* das disciplinas. No primeiro caso, foram identificadas as seguintes falhas:

- O *status* do sistema está sendo exibido como texto puro e não como *links* para as páginas correspondentes. Isso violava as heurísticas, que são “status do sistema” e “controle do usuário e liberdade”; e,
- Botões de controle como “Logout”, “Configurações”, “Sobre” e “Adicionar Disciplina” também violaram as heurísticas “estética e design minimalista” e “reconhecimento

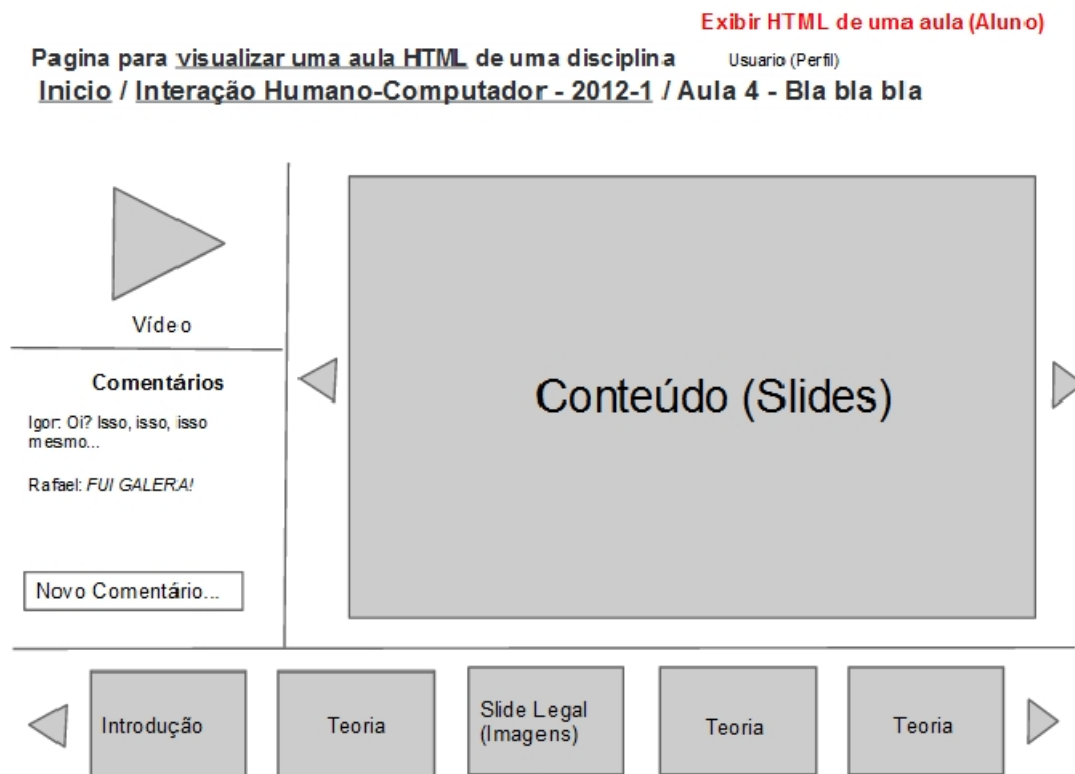


Figura 5.4: Protótipo da tela de apresentação do conteúdo de uma aula capturada.

ao invés da lembrança”, pois não estavam em localizações favoráveis na página e deveriam ganhar novo posicionamento, tanto para otimizar a questão do espaço disponível quanto para favorecer o reconhecimento das funções do sistema pelo usuário, buscando adaptá-lo a ambientes mais familiares.

Na tela de exibição dos *slides* das disciplinas, focou-se novamente na melhor utilização possível do espaço disponível, no intuito de exibir aos usuários as informações mais importantes de maneira clara e de fácil navegação. A heurística “estética e design minimalista” foi novamente mencionada pelo avaliador como um fator a ser melhorado no sistema.

Dessa forma, a avaliação heurística do protótipo do CX foi realizada com sucesso e gerou melhorias significativas, principalmente no que tange fatores como controle do usuário, apresentação da informação, padronização e aparência. Assim, alguns pontos do protótipo da interface foram modificados para contemplar as sugestões recebidas para solucionar os problemas encontrados.

Além da inspeção de usabilidade com especialistas, também foi feito um teste de usabilidade com participação de usuários. Esse teste foi realizado em um ambiente controlado e teve como objetivo encontrar eventuais falhas de *design* na nova interface proposta para o CX. Esse teste contou com a participação de cinco usuários avançados⁴ que apresentavam bom entendimento sobre interfaces.

⁴Alunos da disciplina Interação Humano-Computador do Programa de Pós-Graduação em Ciência da Computação - FACOM/UFU

O método utilizado para a realização do teste é chamado de Método de Avaliação de Comunicabilidade (MAC) adaptado ao domínio educacional por de Oliveira *et al.* (2010).

O MAC requer a participação de usuários em um ambiente controlado – de forma que possam realizar as atividades sem serem interrompidos – e possui como objetivo distinguir e antever problemáticas de *design* por meio da interpretação que o avaliador realiza ao observar a forma que os usuários se comportam e as expressões faciais que deixam transparecer durante a interação com o sistema. Este método caracteriza-se pelo uso da técnica de *Think Aloud* (Nielsen, 1993) e é realizado por meio de três fases: a preparação do teste, a coleta e a análise dos dados.

Esse teste foi preparado para que os usuários executassem três tarefas no sistema:

1. Realizar *login* no sistema e matricular-se em uma disciplina.
2. Acessar os *slides* de uma aula da disciplina matriculada presente na tela inicial do sistema.
3. Acessar uma aula não listada na tela inicial da disciplina.

Ao terminar a execução das atividades propostas, todos os usuários responderam um questionário com perguntas relativas ao sistema avaliado.

Em média, esses usuários executaram todas as tarefas propostas no teste com um tempo médio de 2 minutos. Nenhum deles relatou algum erro gravíssimo na usabilidade proposta e as considerações feitas pelos usuários foram levadas em conta ao projetar a versão final da interface.

Assim sendo, a interface de acesso do CX foi implementada de acordo com o projeto de usabilidade realizado e contou com a utilização de tecnologias Web recentes, como jQuery⁵ e CSS3⁶ (*Cascading Style Sheets*).

A Figura 5.5 mostra a tela inicial do sistema contendo todas as disciplinas matriculadas pelo aluno e a lista das disciplinas ministradas, caso o usuário seja um instrutor. Pode-se notar a existência do botão “Matricular-se em nova turma” do lado esquerdo dessa tela. Esse botão é utilizado para que um usuário se matricule nas disciplinas cadastradas no sistema. Ao clicá-lo, uma *popup* é apresentada para exibir a lista de turmas disponíveis para matrícula. O botão “Adicionar nova turma” é apresentado apenas para os instrutores.

Vale ressaltar que as turmas já matriculadas não são exibidas nessa lista, que é agrupada em quatro níveis: (1) *Área* > (2) *Curso* > (3) *Disciplina* > (4) *Turma*, conforme mostrado na Figura 5.6. Também é possível observar que as turmas podem exigir um código de inscrição para efetuar a matrícula mas elas também podem ser públicas (código de inscrição não exigido). Esse código é informado pelo instrutor no momento que o mesmo realiza o cadastro de uma nova turma.

⁵<http://jquery.com>

⁶<http://www.w3schools.com/css3>

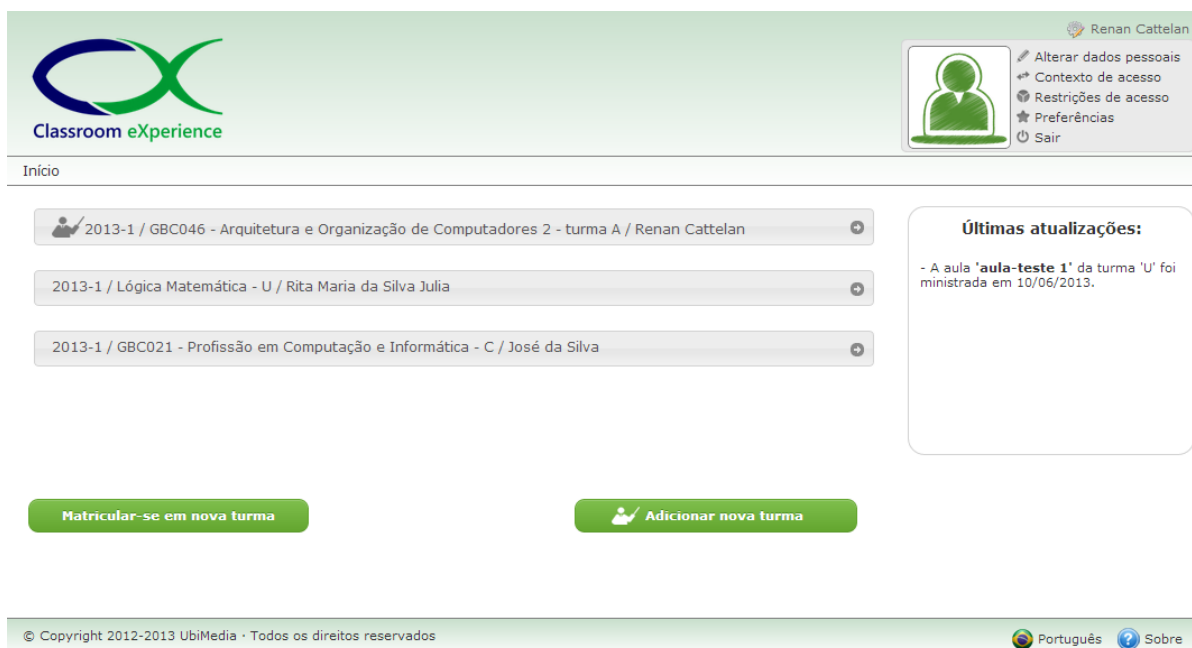


Figura 5.5: Página principal de listagem das disciplinas matriculadas no *Classroom eXperience*.

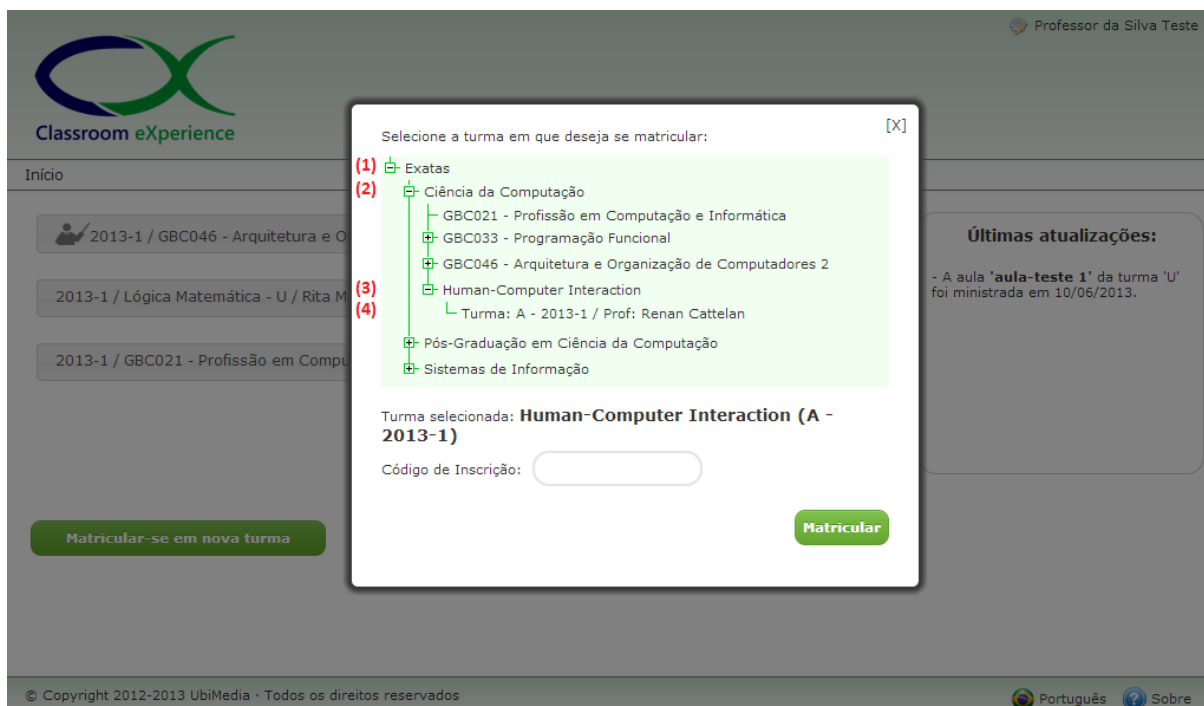


Figura 5.6: *Popup* de exibição das turmas disponíveis para matrícula.

A versão final da apresentação do conteúdo multimídia de um aula no formato HTML pode ser visualizada na Figura 5.7.

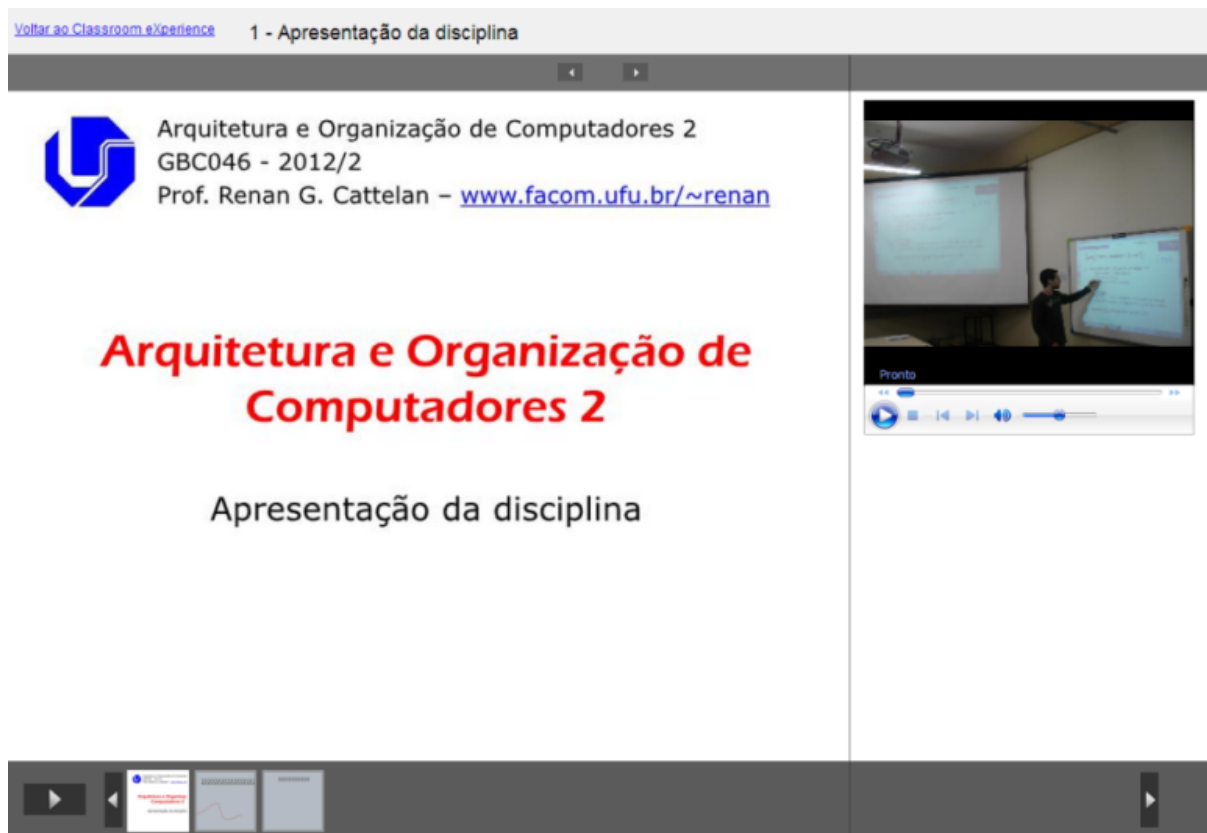


Figura 5.7: Documento HTML apresentando os *slides* e o vídeo de uma aula capturada.

5.4.2 Personalização e Recomendação de Conteúdo Utilizando Contexto, Preferências e Restrições

A Arquitetura de Acesso Contextual (CAA) apresentada no capítulo anterior é utilizada nessa fase para criar uma experiência de acesso personalizada e focada no aluno com o intuito de incentivá-lo a visualizar e interagir com o conteúdo apresentado.

O CX utiliza as 7 dimensões de contexto apresentadas na Figura 4.1. As dimensões *Local*, *Motivo* e *Tempo disponível* são explicitamente informadas pelo usuário enquanto as dimensões *Tipo de dispositivo*, *Velocidade de conexão*, *Resolução de tela* e *Data e hora* são automaticamente inferidas pelo sistema.

Quanto à manipulação de preferências de usuários, o CX utiliza a linguagem CPref-SQL (de Amo e Ribeiro, 2009) que é uma extensão da linguagem SQL construída para dar suporte a consultas baseadas em preferências contextuais e sua sintaxe permite expressar preferências condicionais através de uma abordagem qualitativa.

A linguagem de consultas CPref-SQL é uma extensão do padrão SQL com dois operadores de preferência incluídos: *Select-Best* e *SelectK-Best*. CPref-SQL foi implementada

diretamente no núcleo do processador de consultas do banco de dados PostgreSQL 8.4 (Pereira e de Amo, 2010). Essa abordagem está diretamente relacionada com a possibilidade de recomendação do conteúdo capturado pela CAA, visto que a busca do conteúdo diretamente no banco de dados já retorna um *ranking* dos melhores resultados de acordo com as preferências do usuário e seu contexto de acesso.

As preferências dos usuários são representadas através de *regras de preferência* da forma “SE (um contexto *c* e alguma condição são fornecidos) ENTÃO os valores para o atributo *X* que satisfazem uma condição *S* são *preferíveis* aos valores que não satisfazem essa condição”. Por exemplo, um usuário com pouco tempo disponível para estudos poderia informar que ele prefere visualizar primeiramente as aulas mais importantes e também aquelas mais fáceis (a informação de dificuldade pode ser informada pelo instrutor ao criar a aula na fase de pré-produção).

Assim, a utilização de preferências pessoais no CX visa permitir que os usuários informem suas preferências para que o sistema personalize as consultas realizadas. A utilização de preferências não é obrigatória para o funcionamento do sistema, mas caso não sejam informadas, o sistema apresentará a lista de todas as aulas da disciplina em questão.

Para informar suas preferências pessoais, o aluno pode acessar uma área específica do CX para cadastrá-las. Para cadastrar uma preferência, três informações são exigidas:

1. Um nome arbitrário (rótulo) para a preferência;
2. O nome das tabelas no banco de dados que a preferência está relacionada; e,
3. Uma condição cuja sintaxe deve seguir as especificações da linguagem CPref-SQL (de Amo e Ribeiro, 2009), o que se torna uma limitação do sistema pelo fato de ser informada manualmente.

A Figura 5.8 mostra um formulário Web para cadastrar uma nova preferência pessoal de acordo com os campos (1), (2) e (3), respectivamente. Nesse exemplo, o usuário está informando que ele prefere visualizar as aulas com duração de até 60 minutos ao invés de visualizar as aulas com duração entre 60 e 120 minutos.

O instrutor também pode cadastrar um calendário para cada disciplina com as datas de provas, trabalhos e outras atividades avaliativas. Esse calendário é visualizado pelos alunos na página principal da disciplina. Essa informação pode ser utilizada para recomendar conteúdo. Por exemplo, o aluno não acessa o sistema há uma semana e então realiza uma visita no dia anterior a uma prova agendada; assim, o sistema pode sugerir a ele aquelas aulas marcadas pelo instrutor como relacionadas àquela prova. A visualização do conteúdo recomendado não é obrigatória e o aluno pode listar as aulas de uma disciplina em ordem cronológica ou por título. A Figura 5.9 mostra a página inicial de uma disciplina na visão de um instrutor (na visão de um aluno, os botões cadastrais não são exibidos).

The screenshot shows the 'Classroom eXperience' web interface. At the top, there is a logo and the name 'Luiz Carlos Andrada'. Below the header, a breadcrumb trail reads 'Início > Preferências'. The main section is titled 'Informe os dados abaixo para cadastrar uma nova preferência:'. It contains three form fields, each with a red number in parentheses to its right: (1) 'Nome*:' with the value 'Preferência 1'; (2) 'Tabelas*:' with the value 'aula'; and (3) 'Condições*:' with a complex logical expression: '1 <= duracao <= 60 > 60 < duracao <= [1,2,3,5,6,7,8,9,10,11,12,13,14]'. Below these fields is a green button labeled 'Salvar Preferência'. At the bottom left, there is a note '* Campos obrigatórios.'. The footer contains copyright information '© Copyright 2012-2013 UbiMedia · Todos os direitos reservados', a language selector for 'Português', and a link 'Sobre'.

Figura 5.8: Formulário Web utilizado para cadastrar uma nova preferência pessoal.

Por último, as restrições de apresentação proporcionam a personalização do conteúdo multimídia, influenciando a forma com que o conteúdo é visualizado e quais mídias devem ser apresentadas de acordo com o contexto de acesso do usuário. Por exemplo, se o aluno está usando um *smartphone* na academia, seria melhor apresentar somente o áudio da aula ao invés dos slides ou do vídeo.

Um outro exemplo pode ser observado quando o aluno acessa o sistema com um dispositivo que não possui saída de áudio e ainda o acesso ocorre em um lugar público que, geralmente, contém muitos ruídos. Assim, o áudio da aula poderia ser omitido e a apresentação acontecer apenas com os slides. O conteúdo da aula é formatado através de folhas de estilo que se comportam de acordo com o contexto de acesso do usuário e das restrições de apresentação.

Essas restrições são armazenadas em documentos XML, como aquele apresentado no Fragmento 4.1. Primeiramente o sistema define as restrições padrões que são aplicadas à todos os usuários. À medida em que os usuários utilizam o sistema, eles podem definir suas próprias restrições que serão adicionadas ou substituirão as restrições padrões.

É importante ressaltar que apesar do sistema sugerir um estilo de apresentação espe-

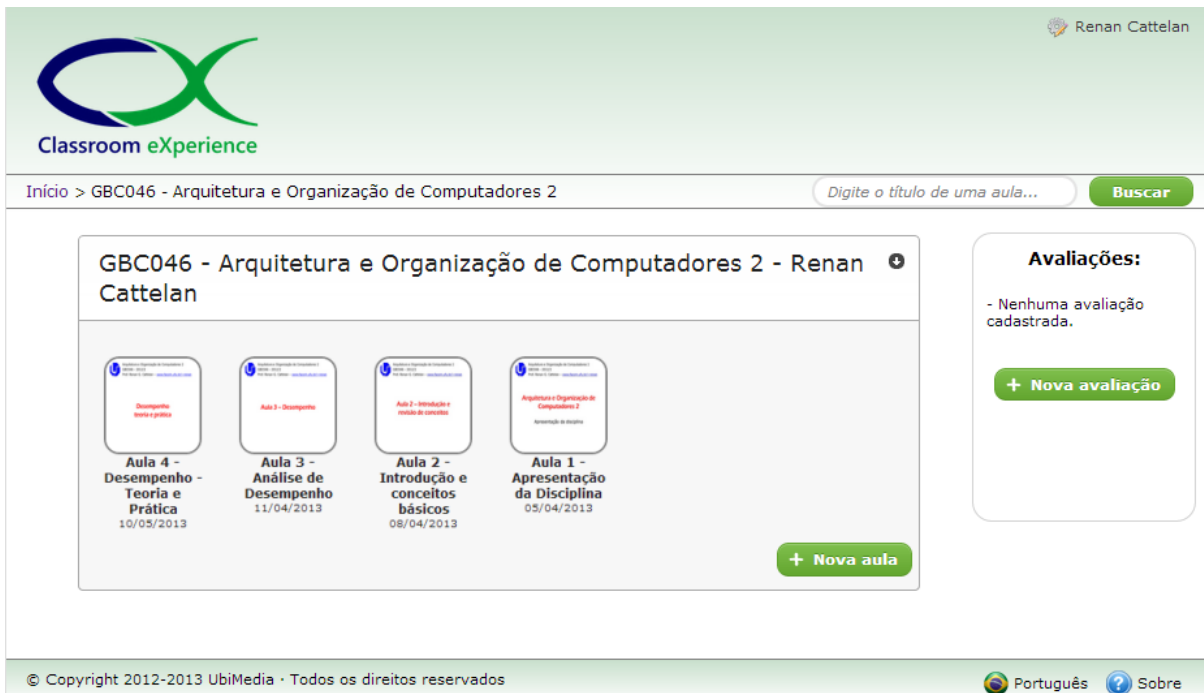


Figura 5.9: Página inicial de uma disciplina cadastrada no *Classroom eXperience*.

cífico, os usuários são livres para escolher as outras formas de apresentação. Na verdade, o sistema apresenta uma lista de documentos de apresentação que são ordenados através de um algoritmo de *ranking*. Atualmente, o CX utiliza um algoritmo de média ponderada que calcula a similaridade entre o contexto do usuário e as restrições de apresentação. No entanto, esse algoritmo pode ser facilmente substituído por outra implementação devido ao baixo acoplamento dos módulos arquiteturais.

Ranking de Apresentação

Cada uma das dimensões de contexto de acesso (*Local*, *Motivo*, *Velocidade de conexão*, *Tempo disponível*, *Dispositivo*, *Data* e *Resolução da tela*) possui uma função de similaridade específica e tais funções são calculadas independentemente umas das outras. Além disso, cada dimensão recebe um peso específico para cada restrição de apresentação. A priori esses pesos são estabelecidos manualmente, mas a arquitetura permite que eles possam ser ajustados de acordo com as estatísticas de acesso.

Cada vez que o aluno acessa o sistema e seu contexto é definido, o sistema gera um *ranking* dos documentos de apresentação que é montado através do cálculo de uma pontuação dada para cada restrição de apresentação existente, de acordo com o contexto de acesso atual.

Essa pontuação é calculada de acordo com a seguinte função:

$$Score(R_i, c) = \sum_{d \in D} w_d * sim_d(R_i(d), c)$$

onde :

R : conjunto de restrições de apresentação

D : dimensões de contexto do usuário

c : instância do contexto do usuário

w_d : peso referente à dimensão de contexto d

sim_d : função de cálculo da similaridade da dimensão d entre a restrição R_i e o contexto c . Seu retorno é sempre entre 0 e 1

Em síntese, a função sim_d aplicada a cada dimensão de contexto é calculada da seguinte maneira:

- *Local, Motivo e Dispositivo*: o cálculo é feito através da comparação entre o valor da restrição e o valor do contexto, ou seja, é uma função booleana que indica se o valor foi escolhido ou não;
- *Velocidade de conexão e Tempo disponível*: o cálculo é feito através de uma função que retorna um número real entre 0 e 1 que indica o quão próximo o valor da instância do contexto está com relação ao valor da restrição;
- *Resolução da tela*: o cálculo é feito através de uma função booleana que resolve a seguinte condição: $(RT_c > RT_r \wedge OP_>) \vee (RT_c \leq RT_r \wedge OP_<)$, onde :
 RT_c : representa o valor da resolução da tela na instância de contexto c
 RT_r : representa o valor da resolução da tela na restrição de apresentação
 $OP_>$: indica que o operador da restrição é ' $>$ '
 $OP_<$: indica que o operador da restrição é ' $<$ '

Com isso, a pontuação atribuída à cada restrição de apresentação é calculada com a soma das funções aplicadas em cada dimensão de contexto contida na restrição, sendo que pesos são dados a essas dimensões para que o resultado final seja normalizado entre 0 e 1. Os pesos dados às dimensões de contexto são personalizáveis de acordo com cada restrição.

O Exemplo 5.1 ilustra o cálculo da função *Score* de uma restrição de apresentação, dado uma instância de contexto arbitrariamente escolhida para deixar claro a sua aplicação.

Exemplo 5.1. Considere a restrição de apresentação R_1 , mostrada na Figura ??, e uma instância de contexto c , cuja definição foi apresentada anteriormente na subseção 4.1. O cálculo da função $Score(R_1, c)$ é mostrado por este exemplo, considerando os seguintes valores:

$$R_1 = \{ \text{velocidade} < 300, \\ \text{tempo} < 30 \\ \text{resolucao} > 800 \times 600$$

motivo = 'revisao rapida'}

c = (1, 1, 'desktop', 800, '1024x768', '2013-07-08 15:00:00', '', '', '')

Neste caso, o cálculo da função $Score(R_1, c)$ acontece em 4 iterações, sendo uma para cada dimensão de contexto da restrição R_1 . Sua execução acontece conforme as seguintes equações:

1. $w_{velocidade} * sim_{velocidade}(R_1(velocidade), c) = 0.2 * sim_{velocidade}(<300, 800)$,
onde: $sim_{velocidade}(<300, 800) = 1$
2. $w_{tempo} * sim_{tempo}(R_1(tempo), c) = 0.2 * sim_{tempo}(<30, '')$,
onde: $sim_{tempo}(<30, '') = 0$
3. $w_{resolucao} * sim_{resolucao}(R_1(resolucao), c) = 0.4 * sim_{resolucao}('<800x600', '1024x768')$,
onde: $sim_{resolucao}('<800x600', '1024x768') = 1$
4. $w_{motivo} * sim_{motivo}(R_1(motivo), c) = 0.2 * sim_{motivo}('revisao rapida', '')$,
onde: $sim_{motivo}('revisao rapida', '') = 0$

Dessa forma, a equação $Score(R_1, c)$ é resolvida pela soma dos resultados de cada uma das quatro equações anteriores:

$$Score(R_1, c) = (1) + (2) + (3) + (4) = 0.2 + 0 + 0.4 + 0 = 0.6$$

Assim, a função $Score(R_1, c)$ assume o valor 0.6, o que a insere no intervalo de classificação $[0,1]$ do *ranking* de apresentação que é ordenado de forma decrescente.

Nesse contexto, o *ranking* dos formatos de apresentação é montado e apresentado a cada usuário por meio de uma *popup* que contém as informações gerais da aula e o respectivo *ranking* em sua parte inferior, conforme mostrado na Figura 5.10.

Para futuras análises estatísticas de acesso, um módulo de *logs* foi implementado para armazenar informações de acesso dos alunos, como data do acesso, *ranking* de apresentação gerado pelo sistema e posição desse *ranking* que o aluno clicou. Em um conjunto de 70 alunos, uma análise prévia mostrou que aproximadamente 71% dos acessos ao sistema aconteceu com um clique na primeira opção do *ranking* gerado para acesso o conteúdo de cada aula.

Ainda com a análise desses *logs*, pode-se observar a quantidade de acesso às aulas ministradas. A Figura 5.11 mostra um gráfico com o número de acesso às aulas ocorrido no decorrer das primeiras 7 semanas de uso do CX. Provavelmente, o pico de acessos apresentado na quarta semana se deu pelo fato de ter havido avaliações marcadas para aquela semana.

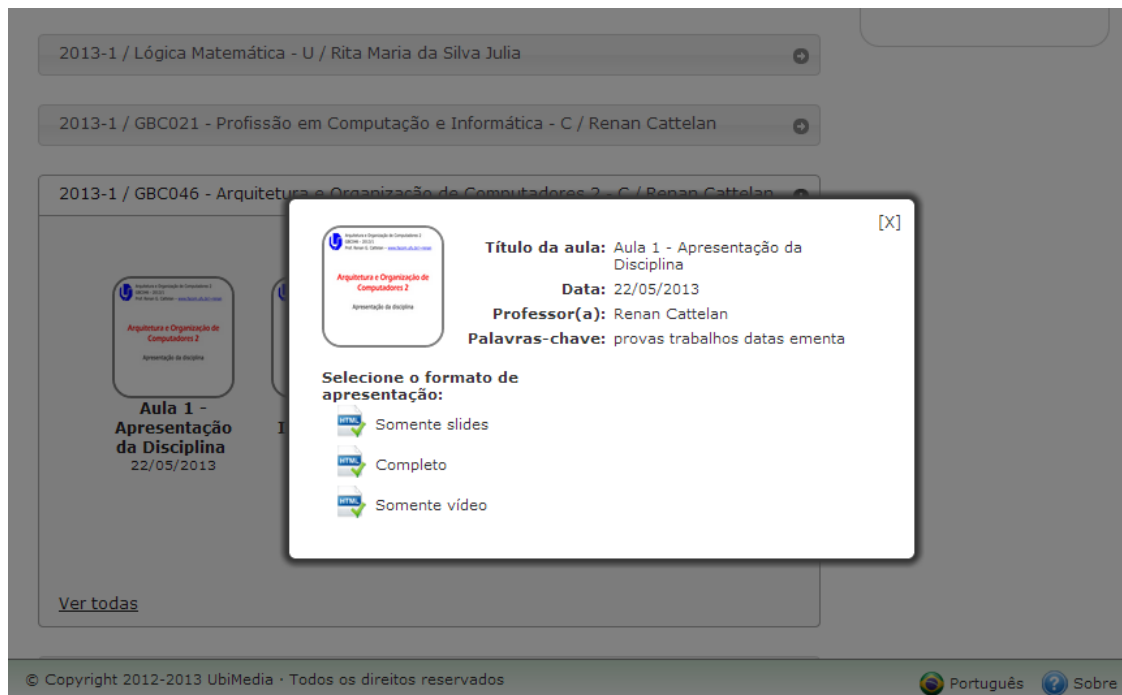


Figura 5.10: *Pop-up* com informações gerais de uma aula e o *ranking* de apresentação (parte inferior).



Figura 5.11: Quantidade de acesso às aulas ministradas.

5.5 Considerações Finais

Neste capítulo, foi apresentado o *Classroom eXperience* (CX), uma plataforma de computação ubíqua utilizada para apoiar as atividades de C&A em ambientes educacionais. Essa plataforma foi utilizada como estudo de caso para implementar e validar o modelo de comunicação (CAL) e a arquitetura de acesso contextual (CAA) que foram propostos.

Ainda, uma nova interface Web, utilizada para acessar o conteúdo capturado, foi projetada e implementada seguindo as boas práticas de projeto de usabilidade e adequada às novas funcionalidades.

Atualmente, 4 instrutores de 5 disciplinas dos cursos de graduação em Ciência da Computação e Sistemas de Informação e de pós-graduação em Ciência da Computação da Faculdade de Computação da Universidade Federal de Uberlândia utilizam o CX durante suas aulas. No total, são atendidos aproximadamente 70 alunos.

Capítulo 6

Trabalhos Relacionados

Existem inúmeros trabalhos na literatura que se relacionam com esta pesquisa. Alguns deles estão focados na captura e outros dão ênfase no acesso, principalmente na recomendação e na personalização de conteúdo multimídia. Assim, eles são apresentados e suas diferenças são apontadas fazendo um paralelo com o trabalho aqui proposto.

A captura de experiências vividas no dia-a-dia é um assunto que tem sido explorado há algum tempo em diversos domínios (Chiu *et al.*, 2001; He *et al.*, 2003; Kunz *et al.*, 2010). Em particular, os sistemas desenvolvidos para ambientes educacionais ganharam espaço e vêm se destacando cada vez mais (AiHua, 2010; Zeng *et al.*, 2010). Possivelmente um dos sistemas mais bem sucedidos desenvolvidos com essa intenção é o eClass (Abowd *et al.*, 1996), também conhecido como *Classroom 2000*, cuja principal funcionalidade é a gravação de aulas a partir de *slides* projetados sobre uma lousa eletrônica, permitindo que anotações sejam feitas pelo instrutor de forma manual. Além dessas anotações, o eClass também captura vídeo a partir de câmeras distribuídas na sala de aula, áudio a partir de um microfone ligado ao corpo do instrutor e também *links* de páginas *web* visitadas durante a aula.

O eClass se assemelha ao trabalho aqui proposto no que tange às atividades de captura, porém a grande diferença se encontra no acesso ao conteúdo capturado, cuja proposta não oferece suporte a recomendação e personalização do conteúdo bem como sua distribuição é feita de forma centralizada. Muitos outros sistemas seguem uma abordagem semelhante (Logan *et al.*, 2012; Pimentel *et al.*, 2007; Shi *et al.*, 2003; Zhang *et al.*, 2008). Nenhum desses sistemas suporta contextualização e personalização de acesso, no entanto.

Algumas outras abordagens também são encontradas na literatura, como é o caso do *Classroom Presenter* (Anderson *et al.*, 2007) que foi desenvolvido pela Universidade de Washington, nos Estados Unidos. Esse sistema permite que instrutores abram apresentações nos formatos PowerPoint 2003, OpenXML ou seu próprio formato binário para

fazer anotações que são sincronizadas com os dispositivos *Tablet PCs* dos alunos. Porém, para que a comunicação entre o instrutor e os alunos ocorra, os dispositivos devem estar internamente na mesma rede e a comunicação ocorre de maneira centralizada, o que pode se tornar um gargalo do sistema.

Outro exemplo bem consolidado de aplicação de C&A para ambientes educacionais é o *Terakoya* (Nishiuchi *et al.*, 2010). Como no *Classroom Presenter*, tal sistema realiza a interação entre instrutores e alunos via *Tablet PCs*, nos quais as anotações feitas pelo instrutor em seu dispositivo são apresentadas aos alunos em tempo real ou mesmo posteriormente. Nesse sistema, as informações são armazenadas em servidores que as replicam entre si, porém os alunos só conseguem acessar o conteúdo se estiverem dentro da mesma rede e o mesmo também vale para os instrutores. Uma das vantagens da abordagem do trabalho aqui proposto é que a comunicação é feita mesmo estando em redes distintas.

Ainda considerando aspectos de comunicação, Barolli e Xhafa apresentam um *middleware* P2P baseado em JXTA para sistemas colaborativos e distribuídos, chamado de *JXTA-Overlay* (Barolli e Xhafa, 2011). Esse *middleware* permite a integração entre dispositivos como sensores e computadores pessoais, provendo transparência e segurança para compartilhar, colaborar e controlar os recursos disponíveis. Apesar dessa abordagem parecer similar com a camada CAL, apresentada nesta dissertação, elas se diferem no fato da CAL não exigir que aplicações que a utilizam conheçam previamente os *peers* da rede para se conectarem. A CAL provê tal abstração de rede e ainda assegura as propriedades de confiabilidade e segurança oferecidas pelo JXTA durante a transferência do conteúdo.

Algumas abordagens possuem características de aplicações síncronas, como é o caso do *Collaboard* (Kunz *et al.*, 2010) que permite a realização de conferências entre colaboradores remotos. Esse sistema captura artefatos de áudio, vídeo e anotações sobre a lousa eletrônica como acontece no CX. Contudo, a transferência do conteúdo capturado se difere nas duas abordagens, pois o *Collaboard* transfere as informações periodicamente visto que ele deve prover aos usuários uma noção de tempo real na captura das informações. Já o *Classroom eXperience* reúne todos os fluxos de mídia capturados na sessão para serem sincronizados e, então, transferidos para seu armazenamento e posterior disponibilização. Outra diferença observada entre as duas abordagens reside nos componentes de software utilizados para captura de áudio e vídeo, que no caso do *Collaboard*, se utiliza o Skype¹ e o CconferenceXP², que são de terceiros. Já o CX utiliza componentes especializados próprios desenvolvidos para a captura de áudio e vídeo.

Semelhante ao *Collaboard*, a aplicação *Tele-Board* (Gumienny *et al.*, 2013) também dá suporte para a colaboração remota entre pessoas, apoiando o trabalho em equipe criativo mesmo se os membros da equipe estiverem localizados em locais distintos. Todas as anotações feitas através de *Tablet PCs*, *smartphones* ou na própria lousa são sincronizadas

¹<http://www.skype.com>

²<http://research.microsoft.com/en-us/projects/conferencexp>

automaticamente com todos os participantes da sessão. A Figura 6.1 mostra a arquitetura do sistema *Tele-Board*. Note que todos os dispositivos conectados utilizam um servidor central de colaboração para sincronizá-los através de mensagens do protocolo XMPP (*Extensible Messaging and Presence Protocol*) para transmissão dos eventos. Existem grandes diferenças entre o *Tele-Board* e o CX. Uma vantagem do *Tele-Board* é que a colaboração entre os usuários é feita de forma síncrona, durante a sessão de captura, e no CX o conteúdo é acessado somente após ser capturado. Isso acontece pelo fato do domínio das duas aplicações ser diferente. Em contrapartida, o modelo de comunicação proposto neste trabalho é descentralizado, o oposto daquele que é utilizado pelo *Tele-Board*.

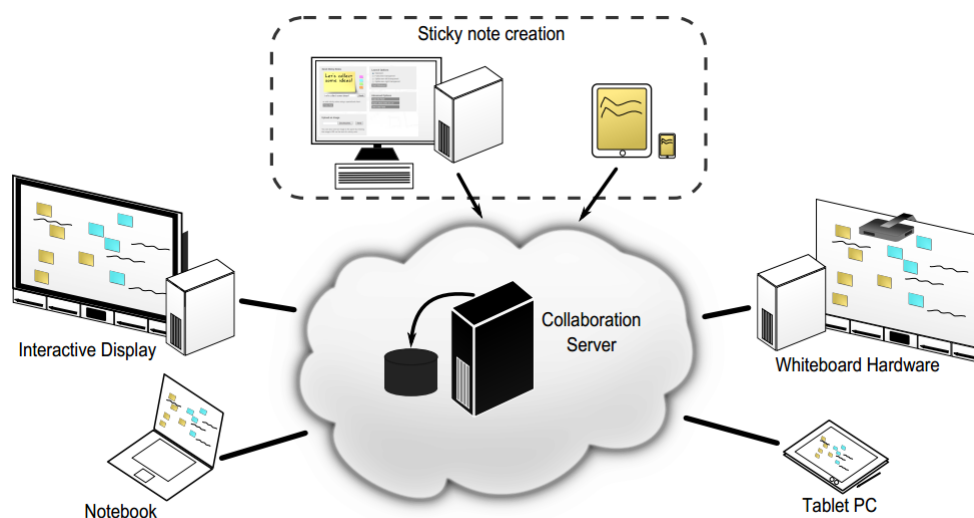


Figura 6.1: Arquitetura do sistema *Tele-Board* (Gumienny *et al.*, 2013).

Voltando para o domínio educacional, as lousas eletrônicas nem sempre são utilizadas. Alguns trabalhos utilizam câmeras de alta resolução para capturar o conteúdo em lousas tradicionais, como é o caso de *ReBoard* (Branham *et al.*, 2010). Esse sistema captura imagens da lousa para disponibilizá-las posteriormente aos alunos via Web ou através de um dispositivo chamado *Chumby*³. Diferentemente do que acontece no CX, o *ReBoard* realiza a captura das imagens através de câmeras instaladas no ambiente e o CX realiza a captura diretamente da lousa eletrônica. Além disso, a fase de acesso ao conteúdo capturado acontece de forma diferente nas duas propostas. No caso do *ReBoard*, o conteúdo é apresentado de forma estática para todos os usuários e pode ser acessado apenas através do *Chumby*. No CX, o conteúdo capturado não permanece estático, ou seja, ele é individualmente personalizado e ainda pode ser estendido ao longo do tempo.

Dickson *et al.* (2008) também utilizam uma abordagem que captura eventos de áudio, vídeo e informações escritas sobre a lousa. Porém, o sistema captura apenas os eventos considerados significativos ao invés da captura completa da sessão, como acontece no CX. Esses eventos são classificados através de um algoritmo que compara a imagem atual

³<http://www.chumby.org>

com imagens anteriores para identificar se houve mudança no conteúdo exibido ou não, indicando a ocorrência de um evento significativo em caso de mudança para que a captura se inicie. Os autores acreditam que esse processo reduz significativamente a quantidade de conteúdo armazenado sem prejudicar o processo de aprendizagem. Neste trabalho, a captura acontece de forma contínua, visto que todo esse conteúdo pode ser posteriormente recomendado e personalizado de acordo com cada aluno.

Especificamente relacionado à parte de acesso, a habilidade de prover mecanismos eficientes que visam selecionar e apresentar o melhor conteúdo aos usuários é um desafio já conhecido pelos sistemas de C&A (Kientz, 2012; Truong e Hayes, 2009). Uma pesquisa precursora em aplicações cientes de contexto se baseou no *Conference Assistant* (Dey *et al.*, 1999), que é uma aplicação na qual os autores fornecem definições operacionais sobre as noções de contexto e, então, apresentam uma arquitetura genérica para a construção de aplicações dessa natureza. Contudo, ela não oferece suporte a preferências pessoais nem restrições de apresentação.

Wang e Wu (2011) apresentam uma abordagem mais robusta para ambientes de aprendizagem que utiliza ciência de contexto e algoritmos de recomendação de conteúdo para ajudar os estudantes durante seu processo de aprendizado. Duas das funcionalidades do sistema proposto por eles se destacam: (1) a utilização de RFID (*Radio-Frequency Identification*) para prover a noção de ciência de contexto e (2) a utilização das especificações dos padrões SCORM (*Sharable Content Object Reference Model*) que permite a reusabilidade dos objetos e dá suporte à multiplataforma. A Figura 6.2 mostra sua arquitetura em alto nível. Em síntese, o sistema de recomendação utiliza um modelo de mineração de regras de associação que analisa todas as preferências dos alunos de acordo com seu perfil de aprendizado e recomenda o conteúdo que melhor se adequa às suas necessidades. Diferentemente deste trabalho, esse sistema não engloba a parte de captura e também não oferece personalização multimídia do conteúdo capturado.

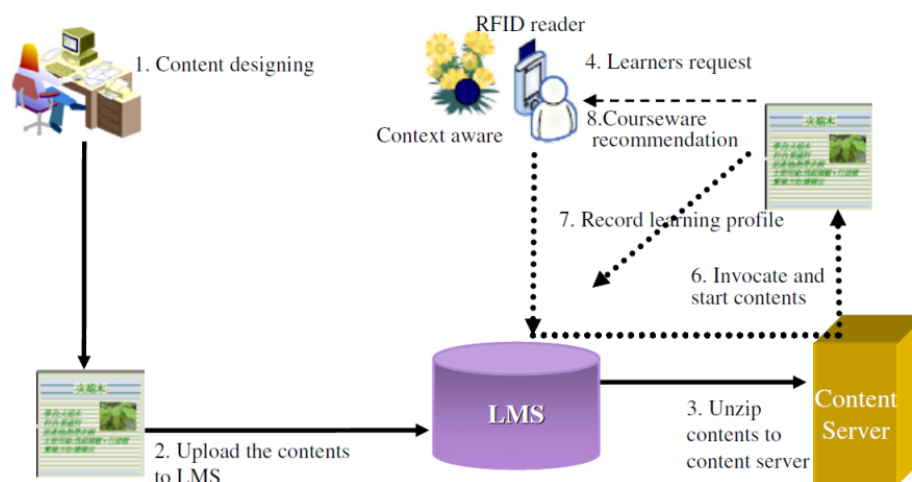


Figura 6.2: Arquitetura do sistema proposto por Wang e Wu (2011).

Ainda na fase de acesso, Miele *et al.* (2009) construíram um modelo de preferências com uma metodologia de refinamento baseado em tuplas e atributos de preferências filtrados sucessivamente. Nesse trabalho, o contexto do usuário é enviado para o seletor de preferências que calcula um *ranking* das tuplas usando um algoritmo “guloso”. Apesar de permitir que a visualização das mídias seja personalizada e apresentada em um *ranking*, esse modelo não utiliza restrições de apresentação. A Figura 6.3 mostra que o fluxo percorrido para personalizar a apresentação do conteúdo é composto por 4 tarefas principais: (1) seleção de preferências ativas, (2) *ranking* de atributos, (3) *ranking* de tuplas e (4) personalização da exibição.

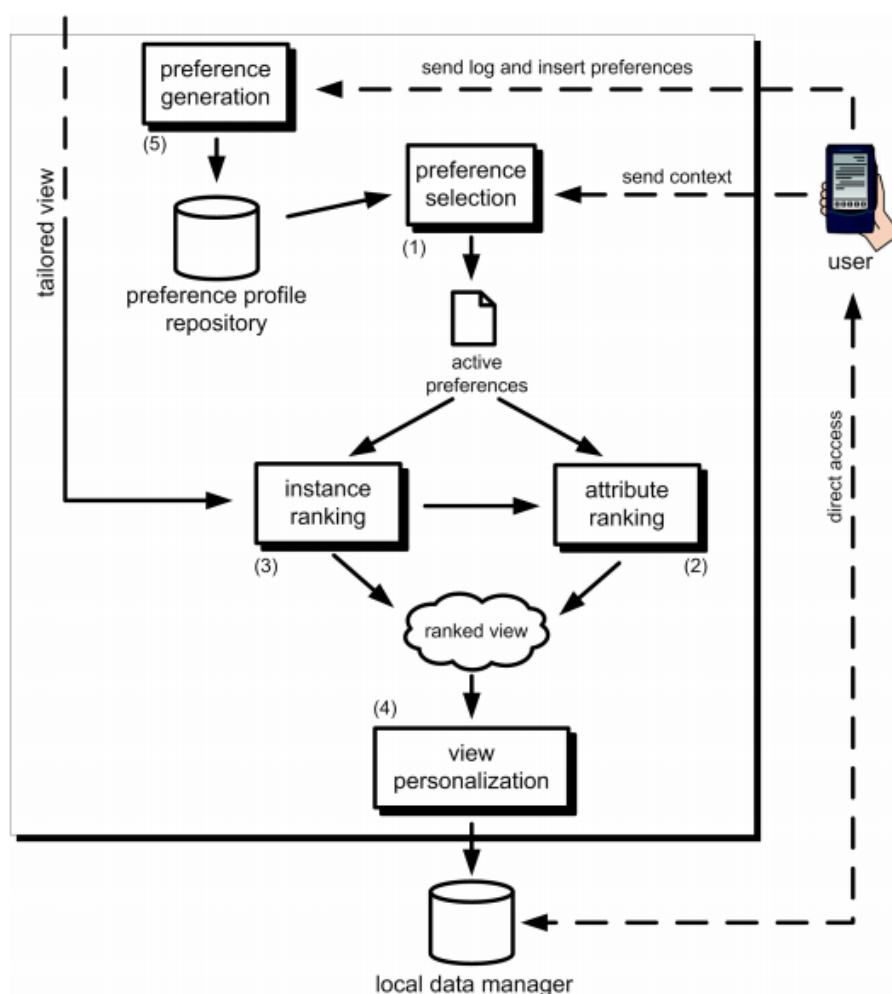


Figura 6.3: Fluxo da personalização de dados contextuais (Miele *et al.*, 2009).

Sugita e Yokota (2012) introduziram a abordagem *User Interface Switching* (UIS), a qual permite a adaptação de conteúdo multimídia de acordo com as aptidões mentais e físicas dos usuários, com a infraestrutura computacionais e qualidade do serviço (QoS) de rede. Essa abordagem se aproxima da utilização de restrições de apresentação vinculadas ao contexto de acesso, porém não são definidas conforme apresentado nessa dissertação. Além disso, essa abordagem não se enquadra em aplicações de C&A.

Lemos *et al.* (2012) apresenta o MMedia2U, um sistema móvel que recomenda fotos baseado no contexto do usuário. Nesse trabalho, o contexto de acesso é modelado como uma ontologia OWL considerando três dimensões de contexto: espacial, social e temporal. A recomendação é baseada em similaridade, porém eles não lidam com restrições de apresentação. Lopes *et al.* (2012) também trabalham de maneira similar, porém com objetivo de adaptar a aplicação de acordo com regras pré-estabelecidas especificamente em cada domínio. Para isso, utiliza uma arquitetura baseada em ontologia e modelos semânticos. Contudo, essa proposta não se enquadra em aplicações de C&A e não oferece suporte para recomendação de conteúdo.

Por fim, Ferreira *et al.* (2012b) apresentam um modelo ubíquo para recomendação de conteúdo educacional. Esse modelo apresenta uma arquitetura descentralizada e é baseada em sistemas multiagentes, dentre eles estão o agente de contexto, os agentes de perfil do usuário e o agente de objetos de aprendizagem. Entretanto, esse modelo também não contempla restrições de apresentação para personalização de conteúdo multimídia.

Com o intuito de sintetizar as informações aqui apresentados, a Tabela 6.1 apresenta os principais trabalhos de acordo com sua área de atuação no âmbito da computação ubíqua: captura e acesso e/ou ciência de contexto. Descendo um nível, esses trabalhos ainda são classificados quanto ao suporte a recomendação e à personalização de conteúdo.

Tabela 6.1: Classificação dos trabalhos relacionados.

Trabalhos	C&A	Ciência de Contexto	Recomendação	Personalização
Chiu <i>et al.</i> (2001)	X	X		X
He <i>et al.</i> (2003)	X			
Abowd <i>et al.</i> (1996)	X	X		
Shi <i>et al.</i> (2003)	X			
Pimentel <i>et al.</i> (2007)	X			
Zhang <i>et al.</i> (2008)	X			
Logan <i>et al.</i> (2012)	X			
Anderson <i>et al.</i> (2007)	X			
Branham <i>et al.</i> (2010)	X			
Dickson <i>et al.</i> (2008)	X			
Dey <i>et al.</i> (1999)		X		
Wang e Wu (2011)		X	X	
Miele <i>et al.</i> (2009)		X	X	X
Lemos <i>et al.</i> (2012)		X	X	
Lopes <i>et al.</i> (2012)		X		X
Ferreira <i>et al.</i> (2012b)		X	X	
Este trabalho	X	X	X	X

Dentre os trabalhos que oferecem suporte a recomendação e a personalização de conteúdo, pode-se refinar ainda mais sua classificação, analisando-os quanto a utilização de contexto de acesso, preferências de usuários e restrições de apresentação, conforme mostrado na Tabela 6.2.

Tabela 6.2: Informações utilizadas para recomendação e personalização de conteúdo.

Trabalhos	Contexto	Preferências pessoais	Restrições de apresentação
Chiu <i>et al.</i> (2001)	X	X	
Wang e Wu (2011)	X	X	
Miele <i>et al.</i> (2009)	X	X	
Lemos <i>et al.</i> (2012)	X	X	
Lopes <i>et al.</i> (2012)	X	X	X
Ferreira <i>et al.</i> (2012b)	X	X	
Este trabalho	X	X	X

Algumas abordagens estão focadas na recomendação de conteúdo enquanto outras focam na personalização da apresentação aos usuários. Essas abordagens podem, ainda, ser combinadas com a utilização de contexto, preferências e restrições no intuito de melhorar ainda mais seu resultado perante a satisfação dos usuários.

Contudo, percebe-se que a abordagem apresentada nesta dissertação se difere de todas as outras em uma visão mais ampla, considerando desde o modelo de comunicação utilizado para a transferência do conteúdo capturado até a utilização de preferências relacionadas com o contexto de acesso dos usuários que, então, geram restrições de apresentação que permitem apresentar o conteúdo multimídia capturado de forma personalizada.

Capítulo 7

Conclusão

A computação ubíqua pode ser frequentemente encontrada no cotidiano das pessoas, visto que a utilização de recursos computacionais e tecnológicos vem sendo amplamente utilizados com o intuito de auxiliar na execução das tarefas diárias.

Especificamente no domínio educacional, os computadores estão sendo utilizados para apoiar os instrutores no processo de ensino e também os alunos ao aprender o conteúdo ensinado pelos instrutores. Com isso, a interatividade entre instrutores e alunos tem aumentado com o uso de aplicações de captura e acesso, autoria multimídia e aplicações colaborativas.

Como já observado, esses tipos de aplicações geram um grande volume de conteúdo que devem ser armazenados e claramente organizados para que sejam exibidos de forma intuitiva aos interessados.

Dessa forma, a abordagem discutida nesta dissertação incluiu um modelo de comunicação baseado na arquitetura *peer-to-peer* (P2P) para armazenar e distribuir o conteúdo multimídia capturado e, ainda, uma arquitetura de acesso contextual (CAA) que dá suporte a recomendação e a personalização de conteúdo utilizando contexto de acesso diretamente ligado com preferências pessoais e restrições de apresentação de forma complementar.

O *Classroom eXperience* foi apresentado como um estudo de caso onde essas propostas puderam ser implementadas como uma prova de conceito real. Trata-se de uma infraestrutura que captura todos os fluxos de mídias disponíveis nas salas de aula instrumentadas para disponibilizá-las posteriormente aos alunos em documentos hipermídia que os integre. Ao utilizarem o sistema, os instrutores podem ministrar suas aulas de forma mais interativa, enquanto os alunos fazem uso de uma ampla base de estudo, com todo o conteúdo automaticamente capturado em sala de aula.

Melhorar as tecnologias que auxiliam o aprendizado é desejável em todo o mundo e

a plataforma apresentada neste trabalho pode ser vista como uma ferramenta poderosa para auxiliar as atividades de ensino e aprendizagem. Além disso, ela pode ser utilizada em diferentes domínios de aplicações de C&A onde o uso de contexto e preferências do usuário é relevante para produzir conteúdo multimídia personalizado.

7.1 Resultados e Contribuições

Esse trabalho traz os seguintes resultados e contribuições técnicas e científicas:

- Definição e implementação de um modelo de comunicação baseado em um protocolo P2P para transferir e replicar o conteúdo multimídia capturado por diferentes tipos de dispositivos instalados em ambientes ubíquos instrumentados (Araújo *et al.*, 2012);
- Elaboração e implementação de uma arquitetura de acesso contextual para recomendar e personalizar o conteúdo multimídia previamente capturado considerando o contexto de acesso do usuário, suas preferências pessoais e as restrições de apresentação. Essa arquitetura foi projetada de modo que seus módulos fossem fracamente acoplados com o intuito de serem facilmente substituídos por outras possíveis implementações que satisfaçam as necessidades de cada aplicação (Araújo *et al.*, 2013b);
- Projeto e implementação da interface de acesso do *Classroom eXperience* baseado em um projeto de usabilidade feito e avaliado pelos usuários; e,
- Desenvolvimento de uma aplicação real que tem sido utilizada pelos instrutores da Faculdade de Computação-UFU para ministrar as aulas de algumas disciplinas dos cursos de graduação em Ciência da Computação e Sistemas de Informação e cursos de Mestrado e Doutorado (Araújo *et al.*, 2013a; Ferreira *et al.*, 2012a).

7.2 Limitações

Algumas limitações da plataforma são encontradas, porém elas não inviabilizam a utilização do sistema por se tratar de aspectos de implementação que podem ser evoluídos:

- A coleta de três das dimensões de contexto de acesso do usuário (Local, Motivo e Tempo disponível) ainda é realizada manualmente;
- O cadastro de preferências pessoais é feito de forma manual e deve obedecer uma sintaxe específica, o que se torna uma tarefa difícil para a maioria dos usuários; e,
- O conteúdo NCL não está sendo distribuído diretamente para *set-top boxes*. Com isso, o download do arquivo com as mídias capturadas para ser utilizado em um emulador ou *set-top box* fica sob responsabilidade do usuário;

7.3 Trabalhos Futuros

As características intrínsecas do sistema apresentado fornecem uma base substancial para pesquisas futuras nas áreas de captura, autoria e apresentação de conteúdo multimídia, recomendação e personalização de conteúdo provenientes de atividades educacionais, etc.

Esta pesquisa vislumbra os seguintes trabalhos futuros:

- Desenvolvimento de técnicas automatizadas para recuperação das preferências dos usuários baseadas em mineração de dados;
- Criação de suporte a anotações colaborativas para enriquecimento do conteúdo já capturado, tanto na *Web* quanto pela TV Digital Interativa;
- Utilização das informações colaborativas realizadas entre os alunos para melhorar a recomendação do conteúdo;
- Estudo de abordagens estatísticas a fim de identificar dentre os artefatos gerados, quais deles possuem maior relevância para os alunos e qual seu real impacto sobre o processo de aprendizagem;
- Avaliação de novos algoritmos para *ranking* dos documentos de apresentação;
- Modelagem do contexto de acesso através de ontologias;
- Adaptação da plataforma para atuar também em ambientes de reuniões, sejam planejadas ou não-planejadas;
- Participação de equipes multidisciplinares na análise do uso da plataforma considerando princípios psicopedagógicos e cognitivos;
- Elaboração de um modelo arquitetural para computação em nuvem a partir da arquitetura existente;
- Distribuição do conteúdo NCL diretamente para *set-top boxes*; e,
- Análise de viabilidade e impacto da migração do modelo de comunicação proposto para utilizar o protocolo de Internet IPv6.

7.4 Considerações Finais

Diferentes aspectos de captura e acesso (C&A) de conteúdo multimídia e ciência de contexto foram abordados neste trabalho. Em especial, a computação ubíqua foi vislumbrada no contexto de ambientes educacionais, onde salas de aula são instrumentadas com projetores multimídia, lousas eletrônicas, microfones, câmeras de vídeo e outros dispositivos computacionais.

Com foco principal na problemática de acesso às informações registradas, primeiramente um modelo de comunicação foi elaborado para armazenar e distribuir o conteúdo multimídia de forma confiável e transparente às aplicações, tornando-o altamente disponível através da replicação feita em *background* e de modo passivo. Em segundo lugar, uma arquitetura de acesso contextual foi projetada para facilitar o acesso ao conteúdo desejado e, ainda, para exibi-lo de forma personalizada aos usuários.

Visando verificar a viabilidade na prática dessa abordagem, as propostas foram implementadas em uma aplicação real (CX) de C&A. Para isso, a usabilidade do CX foi projetada e implementada para acomodar o que foi proposto. Atualmente, essa aplicação está sendo utilizada pelos instrutores e alunos da Faculdade de Computação (FACOM) da Universidade Federal de Uberlândia (UFU), porém pretende-se levá-la aos outros departamentos da UFU, bem como para outras instituições de ensino.

Referências

- Abowd, G. D. (1999). Classroom 2000: an experiment with the instrumentation of a living educational environment. *IBM Systems Journal*, 38(4):508–530.
- Abowd, G. D. e Mynatt, E. D. (2000). Charting Past, Present and Future Research in Ubiquitous Computing. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 7(1):29–58.
- Abowd, G. D., Atkeson, C. G., Feinstein, A., Hmelo, C., Kooper, R., Long, S., Sawhney, N., e Tani, M. (1996). Teaching and Learning as Multimedia Authoring: The Classroom 2000 Project. In *Proceedings of the 4th ACM International Conference on Multimedia*, MULTIMEDIA'96, páginas 187–198, Boston, Massachusetts, USA. ACM.
- Abowd, G. D., Dey, A. K., Brown, P. J., Davies, N., Smith, M., e Steggles, P. (1999). Towards a Better Understanding of Context and Context-Awareness. In *Proceedings of the 1st ACM International Symposium on Handheld and Ubiquitous Computing*, HUC'99, páginas 304–307, Karlsruhe, Germany. Springer-Verlag.
- AiHua, Z. (2010). Study of Ubiquitous Learning Environment Based on Ubiquitous Computing. In *Proceedings of the 3rd IEEE International Conference on Ubi-Media Computing*, U-Media'10, páginas 136–138, Jinhua, China.
- Ali Al-Dahoud, M. W. e Walkowiak, T. (2010). *E-Learning Experiences and Future*. InTech, New York, NY, USA, 1st edition.
- Anderson, R., Anderson, R., Davis, P., Linnell, N., Prince, C., Razmo, V., e Videon, F. (2007). Classroom Presenter: Enhancing Interactive Education with Digital Ink. *Computer*, 40(9):56–61.
- Araújo, R. D., Ferreira, H. N., Rosa, P. F., e Cattelan, R. G. (2012). A Redundancy Information Protocol for P2P Networks in Ubiquitous Computing Environments: Design and Implementation. In *Proceedings of the 11th International Conference on Networks*, ICN'12, páginas 215–220, Saint Gilles, Reunion.
- Araújo, R. D., Brant-Ribeiro, T., Mendonça, I. E., Réu Júnior, E. F., Ferreira, H. N. M., e Cattelan, R. G. (2013a). Collaboration, Adaptation and Personalization of Multimedia Content in Ubiquitous Learning Environments. In *Submetido para o XIX Brazilian Symposium on Multimedia and the Web*, WebMedia'13.

- Araújo, R. D., Brant-Ribeiro, T., Cattelan, R. G., de Amo, S. A., e Ferreira, H. N. M. (2013b). Personalization of Interactive Digital Media in Ubiquitous Educational Environments. In *Aceito pela conferência IEEE International Conference on Systems, Man and Cybernetics*, SMC'13, Manchester, UK. IEEE.
- Barolli, L. e Xhafa, F. (2011). JXTA-Overlay: A P2P Platform for Distributed, Collaborative, and Ubiquitous Computing. *IEEE Transactions on Industrial Electronics*, 58(6):2163–2172.
- Branham, S., Golovchinsky, G., Carter, S., e Biehl, J. T. (2010). Let's Go from the Whiteboard: Supporting Transitions in Work through Whiteboard Capture and Reuse. In *Proceedings of the 28th ACM International Conference on Human Factors in Computing Systems*, CHI'10, páginas 75–84, Atlanta, Georgia, USA. ACM.
- Brotherton, J. A. e Abowd, G. D. (2004). Lessons Learned from eClass: Assessing Automated Capture and Access in the Classroom. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 11(2):121–155.
- Brusilovsky, P. (2001). Adaptive Hypermedia. *User Modeling and User Adapted Interaction*, 11(1):87–110.
- Brusilovsky, P. (2012). Adaptive Technologies for Training and Education. In P. J. Durlach e A. M. Lesgold, editores, *Adaptive Technologies for Training and Education*.
- Business Computing World (2011). Smartphone Geo-Location Tagging Is A Major Security Issue. URL: <http://www.businesscomputingworld.co.uk/smartphone-geo-location-tagging-to-be-a-major-security-issue-in-2012>. Acessado em 30/05/2013.
- Chien, J. (2012). How digital media and Internet is transforming education. In T. Bastiaens e G. Marks, editores, *Proceedings of World Conference on E-Learning in Corporate, Government, Healthcare and Higher Education*, páginas 69–77, Montréal, Quebec, Canada. AACE.
- Chiu, P., Boreczky, J., Girgensohn, A., e Kimber, D. (2001). LiteMinutes: An Internet-Based System for Multimedia Meeting Minutes. In *Proceedings of the 10th International Conference on World Wide Web*, WWW'01, páginas 140–149, Hong Kong, Hong Kong. ACM.
- Cook, D. J. e Das, S. K. (2012). Review: Pervasive computing at scale: Transforming the state of the art. *Pervasive and Mobile Computing*, 8(1):22–35.
- da Silva, W. M., Alvaro, A., Tomas, G. H. R. P., Afonso, R. A., Dias, K. L., e Garcia, V. C. (2013). Smart Cities Software Architectures: A Survey. In *Proceedings of the 28th Annual ACM Symposium on Applied Computing*, SAC'13, páginas 1722–1727, Coimbra, Portugal. ACM.
- de Amo, S. e Ribeiro, M. R. (2009). CPref-SQL: A Query Language Supporting Conditional Preferences. In *Proceedings of the 2009 ACM Symposium on Applied Computing*, SAC'09, páginas 1573–1577, Honolulu, HI, USA.

- de Oliveira, E. R., Prates, R. O., e Werneck, G. A. M. (2010). Considerações sobre Aplicação do Método de Avaliação de Comunicabilidade ao Domínio Educacional. In *Proceedings of the IX Symposium on Human Factors in Computing Systems*, IHC'10, páginas 41–50, Belo Horizonte, MG, BR. Brazilian Computer Society.
- Dey, A., Salber, D., Abowd, G., e Futakawa, M. (1999). The Conference Assistant: combining context-awareness with wearable computing. In *Proceedings of the 3rd IEEE International Symposium on Wearable Computers*, ISWC'99, páginas 21–28, San Francisco, CA, USA.
- Dey, A. K. (2001). Understanding and Using Context. *Personal Ubiquitous Computing*, 5(1):4–7.
- Dickson, P. E., Adrion, W. R., e Hanson, A. R. (2008). Automatic creation of indexed presentations from classroom lectures. *SIGCSE Bulletin*, 40(3):12–16.
- Dix, A., Finlay, J., Abowd, G., e Beale, R. (2004). *Human-Computer Interaction*. Prentice Hall, 3rd edition.
- Ferreira, H., Dias Araujo, R., de Amo, S., e Cattelan, R. (2012a). Classroom Experience: A Platform for Multimedia Capture and Access in Instrumented Educational Environments. In *Proceedings of the Brazilian Symposium on Collaborative Systems*, SBSC'12, páginas 59–64, São Paulo, SP, BR.
- Ferreira, H. N. M. (2012). Captura Multimídia em Ambientes Educacionais Instrumentados: aspectos arquiteturais, modelo de comunicação e interface de acesso contextual. Mestrado em Ciência da Computação, Faculdade de Computação, Universidade Federal de Uberlândia, Uberlândia, MG, BR. Disponível em: http://www.bdtu.ufu.br/tde_busca/processaPesquisa.php?pesqExecutada=1&id=3176.
- Ferreira, L. G. A., Gluz, J. C., e Barbosa, J. L. V. (2012b). Um Modelo Multiagente para Recomendação de Conteúdo Educacional em um Ambiente Ubíquo. In *Anais do 23º Simpósio Brasileiro de Informática na Educação*, SBIE'12, páginas 161–168, Rio de Janeiro, RJ, BR.
- Giliberti, E., Corradini, M., Guaraldi, G., Bertarelli, F., e Genovese, E. (2011). Alternative learning technologies for students with special educational needs. In *Proceedings of the IEEE International Symposium on IT in Medicine and Education*, volume 1 of *ITME'11*, páginas 633–637, Cuangzhou, China.
- Gumienny, R., Gericke, L., Wenzel, M., e Meinel, C. (2013). Supporting Creative Collaboration in Globally Distributed Companies. In *Proceedings of the 16th ACM Conference on Computer Supported Cooperative Work*, CSCW'13, páginas 995–1007, San Antonio, Texas, USA. ACM.
- Hasan, R., Anwar, Z., Yurcik, W., Brumbaugh, L., e Campbell, R. (2005). A Survey of Peer-to-Peer Storage Techniques for Distributed File Systems. In *Proceedings of the International Conference on Information Technology: Coding and Computing*, volume 02 of *ITCC'05*, páginas 205–213, Washington, DC, USA. IEEE Computer Society.
- He, L. W., Liu, Z., e Zhang, Z. (2003). Why take notes? Use the whiteboard capture system. In *Proceedings of the 2003 IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 5 of *ICASSP'03*, páginas 776–779, Hong Kong.

- Holzmann, G. J. (1991). *Design and validation of computer protocols*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA.
- IDC, I. D. C. (2013). Strong Demand for Smartphones and Heated Vendor Competition Characterize the Worldwide Mobile Phone Market at the End of 2012. URL: <http://www.idc.com/getdoc.jsp?containerId=prUS23916413>. Acessado em 16/05/2013.
- IEEE (2002). Draft Standard for Learning Object Metadata. Learning Technology Standards Committee of the IEEE. IEEE LTSC 1484.12.1.
- Kameas, A., Mavrommati, I., Ringas, D., e Wason, P. (2002). eComP: an architecture that supports P2P networking among ubiquitous computing devices. In *Proceedings of the 2nd IEEE International Conference on Peer-to-Peer Computing*, P2P'02, páginas 57–64, Linköping, Sweden.
- Kanso, A., Khendek, F., Mishra, A., e Toeroe, M. (2011). Integrating Legacy Applications for High Availability: A Case Study. In *Proceedings of the IEEE 13th International Symposium on High-Assurance Systems Engineering*, HASE'11, páginas 83–90, Boca Raton, FL, USA. IEEE Computer Society.
- Kientz, J. A. (2012). Embedded capture and access: encouraging recording and reviewing of data in the caregiving domain. *Personal Ubiquitous Computing*, 16(2):209–221.
- Kim, H.-N., Rawashdeh, M., Alghamdi, A., e El Saddik, A. (2012). Folksonomy-based personalized search and ranking in social media services. *Information Systems*, 37(1):61–76.
- Koutrika, G. e Ioannidis, Y. (2010). Personalizing queries based on networks of composite preferences. *ACM Transactions on Database Systems (TODS)*, 35(2):13:1–13:50.
- Kunz, A., Nescher, T., e Kuchler, M. (2010). CollaBoard: A Novel Interactive Electronic Whiteboard for Remote Collaboration with People on Content. In *Proceedings of the IEEE International Conference on Cyberworlds*, CW'10, páginas 430–437, Singapore.
- Lee, M. R. e Chen, T. T. (2009). Trends in Ubiquitous Multimedia Computing. *International Journal*, 4(2):115–124.
- Lemos, F. D., Carmo, R. A., Viana, W., e Andrade, R. M. (2012). Improving photo recommendation with context awareness. In *Proceedings of the 18th Brazilian Symposium on Multimedia and the Web*, WebMedia'12, páginas 321–330, São Paulo, SP, BR. ACM.
- Liang, T.-P., Lai, H.-J., e Ku, Y.-C. (2007). Personalized Content Recommendation and User Satisfaction: Theoretical Synthesis and Empirical Finding. *Journal of Management Information Systems*, 23(3):45–70.
- Lin, M.-R., Lu, S.-H., Ho, T.-H., Lin, P., e Chung, Y.-C. (2007). CFR: a peer-to-peer collaborative file repository system. In *Proceedings of the 2nd International Conference on Advances in Grid and Pervasive Computing*, GPC'07, páginas 100–111, Paris, France, Berlin, Heidelberg. Springer-Verlag.

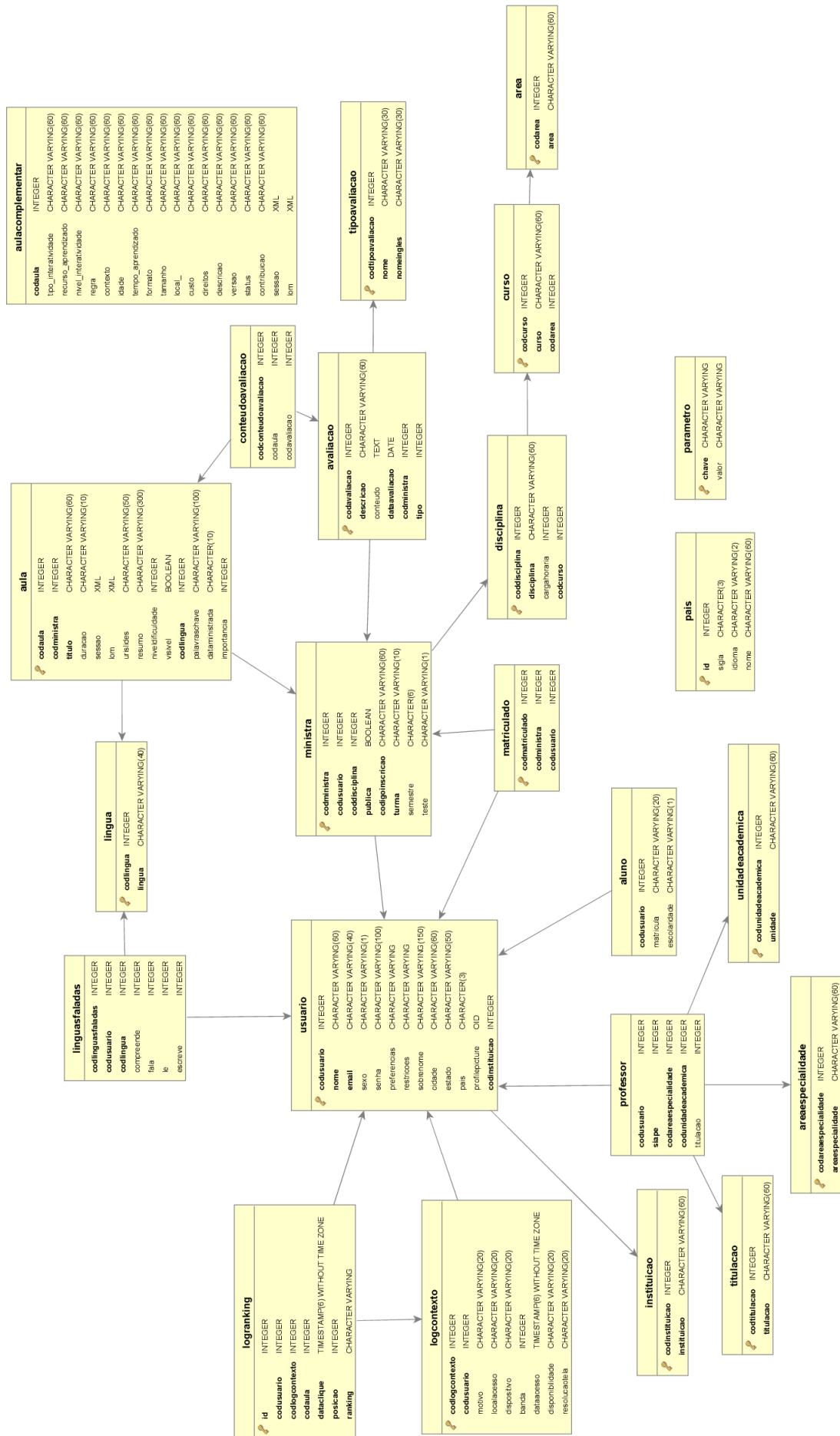
- Logan, G., Greer, J., e McCalla, G. (2012). Thin and Light Video Editing Extensions for Education with Opencast Matterhorn. In *Proceedings of the IEEE International Symposium on Multimedia*, ISM'12, páginas 467–470, Irvine, CA, USA.
- Lopes, J. L., Souza, R. S., Geyer, C. R., Costa, C. A., Barbosa, J. V., Gusmão, M. Z., e Yamin, A. C. (2012). A model for context awareness in Ubicomp. In *Proceedings of the 18th Brazilian Symposium on Multimedia and the Web*, WebMedia'12, páginas 161–168, São Paulo, SP, BR. ACM.
- Miele, A., Quintarelli, E., e Tanca, L. (2009). A methodology for preference-based personalization of contextual data. In *Proceedings of the 12th International Conference on Extending Database Technology: Advances in Database Technology*, EDBT'09, páginas 287–298, Saint Petersburg, Russia. ACM.
- Modahl, M., Agarwalla, B., Abowd, G., Ramachandran, U., e Saponas, T. S. (2004). Toward a standard ubiquitous computing framework. In *Proceedings of the 2nd Workshop on Middleware for Pervasive and ad-hoc Computing*, MPAC'04, páginas 135–139, Toronto, Ontario, Canada. ACM.
- Moos, D. C. e Marroquin, E. (2010). Multimedia, hypermedia, and hypertext: Motivation considered and reconsidered. *Computers in Human Behavior*, 26(3):265–276.
- Nielsen, J. (1993). *Usability Engineering*. Ed. Morgan Kaufmann, San Diego, CA.
- Nishiuchi, Y., Matsuuchi, N., Yamaguchi, T., Shiba, H., Fujiwara, K., Mendori, T., e Shimamura, K. (2010). Enhanced TERA KOYA learning system providing multi-point remote interactive lessons. In *Proceedings of the 8th Asia-Pacific Symposium on Information and Telecommunication Technologies*, APSITT'10, páginas 1–4, Kuching, Malaysia.
- Pereira, F. S. e de Amo, S. (2010). Evaluation of Conditional Preference Queries. *Journal of Information and Data Management (JIDM)*, 1(3):1–16.
- Perich, F., Joshi, A., Finin, T., e Yesha, Y. (2004). On Data Management in Pervasive Computing Environments. *IEEE Transactions on Knowledge and Data Engineering*, 16(5):621–634.
- Pimentel, M., Cattelan, R., e Baldochi, L. (2007). Prototyping Applications to Document Human Experiences. *Pervasive Computing, IEEE*, 6(2):93–100.
- Pimentel, M. d. G. C., Ishiguro, Y., Kerimbaev, B., Abowd, G. D., e Guzdial, M. (2001). Supporting Educational Activities through Dynamic Web Interfaces. 13:353–374.
- Richter, H. A., Abowd, G. D., Geyer, W., Fuchs, L., Daijavad, S., e Poltrock, S. E. (2001). Integrating Meeting Capture within a Collaborative Team Environment. In *Proceedings of the 3rd International Conference on Ubiquitous Computing*, UbiComp'01, páginas 123–138, Atlanta, Georgia, USA. Springer-Verlag.
- Schilit, B. e Theimer, M. (1994). Disseminating active map information to mobile hosts. *IEEE Network*, 8(5):22–32.

- Settle, A., Dettori, L., e Davidson, M. J. (2011). Does lecture capture make a difference for students in traditional classrooms? In *Proceedings of the 16th Annual Joint Conference on Innovation and Technology in Computer Science Education*, ITiCSE'11, páginas 78–82, Darmstadt, Germany. ACM.
- Shi, Y., Xie, W., Xu, G., Shi, R., Chen, E., Mao, Y., e Liu, F. (2003). The smart classroom: merging technologies for seamless tele-education. *IEEE Pervasive Computing*, 2:47–55.
- Silva, H. V. O., Rodrigues, R. F., Soares, L. F. G., e Muchaluat Saade, D. C. (2004). NCL 2.0: integrating new concepts to XML modular languages. In *Proceedings of the 2004 ACM Symposium on Document Engineering*, DocEng'04, páginas 188–197, Milwaukee, Wisconsin, USA. ACM.
- Stefanidis, K., Koutrika, G., e Pitoura, E. (2011). A survey on representation, composition and application of preferences in database systems. *ACM Transactions on Database Systems (TODS)*, 36(3):19:1–19:45.
- Sugita, K. e Yokota, M. (2012). Development of a user interface aware of users' skills and computer environments. In *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*, SMC'12, páginas 2953–2957, Seoul, South Korea.
- Sun Microsystems, I. (2007). Project JXTA v2.5: Java Programmer's Guide.
- Tang, L., Yu, Z., Zhou, X., Wang, H., e Becker, C. (2011). Supporting rapid design and evaluation of pervasive applications: challenges and solutions. *Personal Ubiquitous Computing*, 15(3):253–269.
- Truong, K. N. (2005). *Inca: an infrastructure for capture and access. supporting the generation, preservation and use of memories from everyday life*. Doutorado em Ciência da Computação, Atlanta, GA, USA. Disponível em: <http://khaitruong.com/publications/Dissertation-2005.pdf>. Acesso em: 10/12/2011.
- Truong, K. N. e Hayes, G. R. (2009). *Ubiquitous Computing for Capture and Access*. Now Publishers Inc., Hanover, MA, USA.
- Truong, K. N., Abowd, G. D., e Brotherton, J. A. (1999). Personalizing the capture of public experiences. In *Proceedings of the 12th Annual ACM Symposium on User Interface Software and Technology*, UIST'99, páginas 121–130, Asheville, North Carolina, USA. ACM.
- Truong, K. N., Abowd, G. D., e Brotherton, J. A. (2001). Who, What, When, Where, How: Design Issues of Capture & Access Applications. In *Proceedings of the 3rd International Conference on Ubiquitous Computing*, UbiComp'01, páginas 209–224, Atlanta, Georgia, USA. Springer-Verlag.
- Wang, S.-L. e Wu, C.-Y. (2011). Application of context-aware and personalized recommendation to implement an adaptive ubiquitous learning system. *Expert Systems with Applications: An International Journal*, 38(9):10831–10838.
- Wazlawick, R. S. (2009). *Metodologia de Pesquisa para Ciência da Computação*. Elsevier, Rio de Janeiro, RJ, BR.

- Weiser, M. (1991). The Computer for the 21st Century. *Scientific American*, 265(3):66–75.
- Weiser, M. (1993). Some computer science issues in ubiquitous computing. *Communications of the ACM*, 36(7):75–84.
- Weiser, M. e Brown, J. S. (1996). Beyond calculation. Capítulo The coming age of calm technology, páginas 75–85. Copernicus, New York, NY, USA.
- Wigdor, D. e Wixon, D. (2011). *Brave NUI World: Designing Natural User Interfaces for Touch and Gesture*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1st edition.
- Xie, W., Zhang, Z., Lu, C., Wang, Y., e Shen, R. (2010). PPClass - a classroom lecture broadcast platform based on P2P streaming technology. In *Proceedings of the 4th International Conference on Distance Learning and Education*, ICDLE'10, páginas 199–203, San Juan, PR.
- Zeng, L., Zuo, M., e Lu, X. (2010). Research into application of interactive whiteboard to interactive educational mode. In *Proceedings of the International Conference on Computational Intelligence and Software Engineering*, CiSE'10, páginas 1–4, Wuhan, China.
- Zhang, C., Rui, Y., Crawford, J., e He, L.-W. (2008). An automated end-to-end lecture capture and broadcasting system. *ACM Transactions on Multimedia Computing, Communications and Applications (TOMCCAP)*, 4(1):6:1–6:23.
- Zhao, X., Wan, X., e Okamoto, T. (2010). Adaptive Content Delivery in Ubiquitous Learning Environment. In *Proceedings of the 6th IEEE International Conference on Wireless, Mobile and Ubiquitous Technologies in Education*, WMUTE'10, páginas 19–26, Kaohsiung, China.
- Zhong-xian, W., Jiang, Z., e Wen-Nai, W. (2010). The Protocol Design and Implementation of a JXTA Based P2P-SIP System. In *Proceedings of the International Conference on Internet Technology and Applications*, iTAP'10, páginas 1–4, Wuhan, China.

Apêndice A

Diagrama Entidade-Relacionamento (DER) da base de dados do CX



Apêndice B

Restrições de Apresentação Utilizadas

```
<?xml version="1.0" encoding="iso-8859-1" ?>
<constraints>
  <rule>
    <antecedent>
      <condition att="connection_speed" op="!" val="300" weight="0.2"></condition>
      <condition att="available_time" op="!" val="30" weight="0.2"></condition>
      <condition att="screen_resolution" op=">" val="800x600" weight="0.4"></condition>
      <condition att="motive" op="=" val="quick_review" weight="0.2"></condition>
    </antecedent>
    <consequent>
      <result name="Somente_slides" val="xhtml_slides.xml"></result>
    </consequent>
  </rule>
  <rule>
    <antecedent>
      <condition att="device" op="=" val="desktop" weight="0.3"></condition>
      <condition att="connection_speed" op=">" val="300" weight="0.1"></condition>
      <condition att="available_time" op="!" val="30" weight="0.4"></condition>
      <condition att="screen_resolution" op=">" val="800x600" weight="0.3"></condition>
    </antecedent>
    <consequent>
      <result name="Somente_slides" val="xhtml_slides.xml"></result>
    </consequent>
  </rule>
  <rule>
    <antecedent>
      <condition att="device" op="=" val="desktop" weight="0.1"></condition>
      <condition att="connection_speed" op=">" val="600" weight="0.25"></condition>
      <condition att="available_time" op=">" val="60" weight="0.25"></condition>
      <condition att="screen_resolution" op=">" val="800x600" weight="0.1"></condition>
      <condition att="place" op="=" val="home" weight="0.2"></condition>
      <condition att="motive" op="=" val="lost_lesson" weight="0.1"></condition>
    </antecedent>
    <consequent>
      <result name="Completo" val="xhtml_completo.xml"></result>
    </consequent>
  </rule>
  <rule>
    <antecedent>
      <condition att="device" op="=" val="smartphone" weight="0.1"></condition>
      <condition att="connection_speed" op=">" val="600" weight="0.3"></condition>
```

```
<condition att="available_time" op=">" val="60" weight="0.3"></condition>
<condition att="place" op="=" val="home" weight="0.2"></condition>
<condition att="motive" op="=" val="lost_lesson" weight="0.1"></condition>
</antecedent>
<consequent>
  <result name="Somente_video" val="xhtml_video.xml"></result>
</consequent>
</rule>
</constraints>
```


Anexo A

Estrutura do arquivo *Learning Object Metadata* (LOM) armazenado

```
<?xml version="1.0" encoding="iso-8859-1" ?>
<lom id="1" type="0">
  <general>
    <id>217</id>
    <title>Aula 1</title>
    <language>Portugues</language>
    <description>Aula introdutoria da disciplina</description>
    <keywords>Ementa, Introducao</keywords>
  </general>
  <educational>
    <interactivity_type>Lousa</interactivity_type>
    <learning_resource_type></learning_resource_type>
    <interactivity_level>1</interactivity_level>
    <end_user_rule>R1</end_user_rule>
    <context>C1, C2, C3</context>
    <age_range>20</age_range>
    <difficulty>2</difficulty>
    <learning_time></learning_time>
    <description></description>
    <language>Portugues</language>
  </educational>
  <technical>
    <format>pdf</format>
    <size></size>
    <location>Sala 126</location>
    <duration>100</duration>
  </technical>
  <rights>
    <cost></cost>
    <copyright></copyright>
    <description></description>
  </rights>
  <life_cicle>
    <version></version>
    <status></status>
    <contribute></contribute>
  </life_cicle>
</lom>
```