

UNIVERSIDADE FEDERAL DE ALAGOAS  
INSTITUTO DE COMPUTAÇÃO  
PROGRAMA DE PÓS-GRADUAÇÃO EM MODELAGEM COMPUTACIONAL DE  
CONHECIMENTO

HEMILIS JOYSE BARBOSA ROCHA

**Um Modelo de Referência para Ambientes Virtuais de  
Aprendizagem na Web: Aproximando as Perspectivas do  
Autor e do Desenvolvedor**

Maceió, junho de 2012

HEMILIS JOYSE BARBOSA ROCHA

**Um Modelo de Referência para Ambientes Virtuais de  
Aprendizagem na Web: Aproximando as Perspectivas do  
Autor e do Desenvolvedor**

Dissertação de mestrado apresentada ao Programa  
de Pós-Graduação em Modelagem Computacional  
de Conhecimento do Instituto de Computação da  
Universidade Federal de Alagoas.

Orientadores:

Evandro de Barros Costa

Patrick Henrique da Silva Brito

Maceió, julho de 2012

**Catálogo na fonte**  
**Universidade Federal de Alagoas**  
**Biblioteca Central**  
**Divisão de Tratamento Técnico**  
**Bibliotecária Responsável: Helena Cristina Pimentel do Vale**

R672e Rocha, Hemilis Joyse Barbosa.  
Um modelo de referência para ambientes virtuais de aprendizagem na web :  
aproximando as perspectivas do autor e do desenvolvedor / Hemilis Joyse Barbosa  
Rocha. – 2012.  
137 f. : il.

Orientador: Evandro de Barros Costa.  
Co-Orientador: Patrik Henrique da Silva Brito.  
Dissertação (mestrado em Modelagem Computacional de Conhecimento) –  
Universidade Federal de Alagoas. Instituto de Computação. Maceió, 2012.

Bibliografia: f. 126-131.  
Apêndices: 132-137.

1. Ambientes virtuais de aprendizagem – Usabilidade. 2. Mapa conceitual.  
3. Arquitetura de software. I. Título.

CDU: 004.4



Membros da Comissão Julgadora da Dissertação de Mestrado de Hemilis Joyse Barbosa Rocha, intitulada: "Um modelo de referência para ambientes virtuais na web: aproximando as perspectivas do autor e do desenvolvedor", apresentada ao Programa de Pós-Graduação em Modelagem Computacional de Conhecimento da Universidade Federal de Alagoas em 02 de julho de 2012, às 10h00min, na sala de aula do Mestrado em Modelagem Computacional de Conhecimento.

COMISSÃO JULGADORA

**Prof. Dr. Evandro de Barros Costa**

UFAL – Instituto de Computação

Orientador

**Prof. Dr. Patrick Henrique da Silva Brito**

UFAL – Instituto de Computação

Orientador

**Prof. Dr. Fábio Paraguaçu Duarte da Costa**

UFAL – Instituto de Computação

Examinador

**Prof. Dr. Luis Paulo Leopoldo Mercado**

UFAL – Centro de Educação

Examinador

**Prof. Dr. Sérgio Crêspo Coelho da Silva Pinto**

Unisinos – Centro de Ciências Exatas e Tecnológicas

Examinador

## **Dedicatória**

Dedico este trabalho aos meus pais.

## **Agradecimentos**

Agradeço imensamente a Deus, por tudo.

Agradeço muitíssimo aos meus pais, Maria Selma e José Fontes, por todo amor e força que serviram como suicídios para a realização deste trabalho.

Agradeço a meus irmãos, Helton e John, e toda minha família.

Agradeço ao meu orientador professor Doutor Evandro de Barros Costa por ter me dado à oportunidade salvar o meu mestrado, com seus conselhos, preocupações e entusiasmo.

Agradeço ao meu outro orientador Patrick Henrique Brito, por toda ajuda e paciência.

Agradeço a Tiago Eloy, pelos cuidados e por dividir comigo sua experiência de conclusão de mestrado.

Agradeço a Priscylla e aos demais membros do GRAW.

Agradeço a todos os meus amigos e, em especial, a Fernanda Josirene.

Agradeço a todos os professores e estudantes que participaram das entrevistas que serviram de base para a realização deste trabalho.

Agradeço a Rafael Henrique, por toda a ajuda.

E por fim, agradeço àqueles que não acreditaram na minha capacidade, pois serviu como um incentivo a mais.

## Resumo

Muitas plataformas de Ambientes Virtuais de Aprendizagem têm sido desenvolvidas e disponibilizadas, tendo com isso atraído uma quantidade significativa e diversificada de usuários. Neste contexto, além da diversidade de usuários, ampliam-se as possibilidades tecnológicas, possibilitando-se a viabilização de diferentes propostas pedagógicas, eventualmente demandadas por esse público. Apesar disso, observam-se lacunas importantes no cenário de desenvolvimento ou mesmo customização dos AVAS para atender a requisitos específicos para um determinado uso, isso conduz a dificuldades apresentadas aos autores e, alguns casos, para os próprios desenvolvedores. Assim, investiu-se nesta dissertação em uma possível resposta para amenizar parte desses problemas, focalizando na definição de um modelo de referência que conta com um mapa conceitual destinado a auxiliar educadores a escolher serviços de um AVA que atenda suas necessidades e uma arquitetura de referência que se presta a ajudar os desenvolvedores a implementar as demandas dos usuários. Ademais, desenvolveu-se um sistema de recomendação baseado em conhecimento que visa aproximar a visão do educador da visão do desenvolvedor, mapeando as escolhas do educador, em o que o desenvolvedor precisa para desenvolvê-lo. Com isso, buscou-se oferecer um arcabouço adequado para realizar avaliações e comparações entre AVAs, além de servir para orientar e viabilizar o desenvolvimento de novos ambientes ou ainda customização de soluções atendendo demandas específicas de educadores. Uma avaliação para essa abordagem proposta foi feita por meio de um estudo de caso que serviu para explorar os recursos dos três principais componentes desenvolvidos, obtendo-se um primeiro resultado satisfatório, ainda que bem preliminar, sobre o trabalho aqui apresentado.

**Palavras chaves:** Ambientes virtuais de aprendizagem – Usabilidade. Mapa conceitual. Arquitetura de Software.

## **Abstract**

Many platforms Virtual Learning Environments have been developed and made available, and thereby attracted a significant amount of users and diverse. In this context, in addition to the diversity of users, expand the possibilities of technology, enabling the feasibility of different pedagogical eventually demanded by the public. Nevertheless, there are important gaps in the scenario development or customization of AVAS to meet specific requirements for a particular use, this leads to difficulties presented by the authors and, some cases, the developers themselves. So invested in this dissertation on a possible answer to alleviate some of these problems, focusing on defining a reference model that has a conceptual map designed to help educators choose the services of an AVA that meets your needs and a reference architecture that lends itself to help developers implement the demands of users. In addition, we developed a recommendation system based on knowledge that aims to bring the vision of the developer's vision educator, the educator mapping choices in what the developer needs to develop it. Therefore, we sought to provide a framework suitable to perform evaluations and comparisons between AVAs, and serves to guide and facilitate the development of new environments or customized solutions meeting the specific demands of educators. An assessment for the proposed approach was made through a case study that was used to explore the features of the three main components developed, obtaining a first satisfactory results, although very preliminary, on the work presented here.

**Keywords:** Virtual learning environments - Usability. Conceptual map. Software Architecture.

## Lista de Figuras

Figura 1 - Modelo conceitual baseado em componentes (CRESPO et al, 1998, p. 3) .....	25
Figura 2 – Arquitetura em camadas lógicas (SWEENEY, 2008, p. 62).....	27
Figura 3 Mapa Conceitual do Ambiente.....	30
Figura 4 - Mapa Conceitual Geral do Ambiente Completo – Parte 1 .....	31
Figura 5 - Mapa Conceitual Geral do Ambiente Completo – Parte 2 .....	32
Figura 6 – O Conceito Atores.....	33
Figura 7 - O Conceito Interface.....	34
Figura 8 – O Conceito “Cursos”.....	35
Figura 9 - O Conceito “Percepção do Ambiente” .....	36
Figura 10 - O Conceito “Customização do Ambiente” .....	37
Figura 11 - O Conceito “Internacionalização” .....	38
Figura 12 - O Conceito “Serviços” - parte 1 .....	40
Figura 13 - O Conceito “Serviços” - parte 2 .....	40
Figura 14 - O Conceito Grupos .....	42
Figura 15 - O Mapa Conceitual para o professor - parte 1 .....	44
Figura 16 - O Mapa Conceitual para o professor - parte 2.....	45
Figura 17 - O Mapa Conceitual para o aluno - parte 1 .....	52
Figura 18 - O Mapa Conceitual para o aluno - parte 2.....	53
Figura 19 - Mapa Conceitual do Ambiente Amadeus - parte 2.....	59
Figura 20 - Mapa Conceitual do Ambiente Moodle – parte 1.....	64
Figura 21 - Mapa Conceitual do Ambiente Moodle – parte 2.....	65
Figura 22 - Mapa Conceitual do Ambiente Claroline – parte 1 .....	71
Figura 23 - Mapa Conceitual do Ambiente Claroline – parte 2 .....	72
Figura 24 -Mapa Conceitual do Ambiente Eureka - parte 1.....	77
Figura 25 - Mapa Conceitual do Ambiente Eureka – parte 2.....	78
Figura 26 - Mapa Conceitual do Ambiente Sakai - parte 1 .....	83
Figura 27 - Mapa Conceitual do Ambiente Sakai - parte 2.....	84
Figura 28 - Mapa Conceitual do Ambiente Tidia Ae - parte 1.....	89
Figura 29 - Mapa Conceitual do Ambiente Tidia Ae - parte 2.....	90
Figura 30 - Processo utilizado para projetar a arquitetura de referência.....	102
Figura 31 - Visão Geral da Arquitetura de Referência Proposta.....	104
Figura 32 - Diagrama de Componentes - parte 1.....	112
Figura 33 - Diagrama de Componente - parte 2 .....	113

Figura 34 - Sistema de Recomendação Tela 2 .....	116
Figura 35 - Sistema de Recomendação Tela 3 .....	116
Figura 36 - Sistema de Recomendação Tela 4 .....	117
Figura 37 - Sistema de Recomendação Tela 2 .....	117
Figura 38 - Sistema de Recomendação Tela 3 .....	118
Figura 39 - Sistema de Recomendação Tela 4 .....	118
Figura 40 - Base de Conhecimento .....	118
Figura 41 - Visão Geral da Arquitetura do Sakai .....	120
Figura 42 - Arquitetura Recomendada para o Sakai.....	122
Figura 43 - Diagrama de Componentes Sakai .....	123

## **Lista de Quadros**

Tabela 1 - Análise Comparativa entre os Ambientes .....	95
Tabela 2 - Mapeamento dos requisitos não-funcionais nos estilos .....	108
Tabela 3 - Interfaces Providas pelos Componentes da Infraestrutura (Mapa Conceitual) .....	110
Tabela 4 - Tabela 3 - Interfaces Providas pelos Componentes da Aplicação (Mapa Conceitual) .....	110
Tabela 5 - Mapeamento entre os principais elementos da arquitetura de referência e do <i>Sakai</i> .....	120

# Conteúdo

1. INTRODUÇÃO .....	15
1.1. Contexto, Problemática e Motivação da Pesquisa.....	15
1.2. Objetivos.....	19
1.3. Relevância .....	19
1.4. Aspectos Metodológicos .....	20
1.5. Estrutura da Dissertação .....	22
2. FUNDAMENTAÇÃO TEÓRICA .....	23
2.1. Aspectos de Engenharia de Software .....	23
2.2. Sistemas Baseados em Conhecimento.....	24
3. TRABALHOS RELACIONADOS.....	25
3.1. Perspectivas de um modelo conceitual.....	25
3.2. Perspectivas de uma Arquitetura .....	26
4. O MAPA CONCEITUAL PARA AVAS .....	28
4.1. Metodologia.....	28
4.2. O Mapa Conceitual Geral.....	29
4.2.1. Conceito “atores” .....	33
4.2.2. Conceito “interface” .....	33
4.2.3. Conceito “cursos” .....	34
4.2.4. Conceito “percepção do ambiente”.....	35
4.2.5. Conceito “customização do ambiente” .....	36
4.2.6. Conceito “portabilidade para dispositivos” .....	37
4.2.7. Conceito “internacionalização” .....	38
4.2.8. Conceito “serviços” .....	38
4.2.9. Conceito “grupos”.....	40
4.3. O Mapa Conceitual para o Ator Professor.....	42
4.4. O Mapa Conceitual para o Ator Aluno.....	50
5. AVALIAÇÃO DE AMBIENTES VIRTUAIS DE APRENDIZAGEM .....	58
5.1. Amadeus .....	58
5.1.1. Conceito “atores” no Amadeus.....	59
5.1.2. Conceito “interface” no Amadeus .....	60
5.1.3. Conceito “cursos” no Amadeus .....	60
5.1.4. Conceito “percepção do ambiente” no Amadeus .....	61

5.1.5.	Conceito “customização do ambiente” no Amadeus.....	61
5.1.6.	Conceito “portabilidade para dispositivos” no Amadeus .....	61
5.1.7.	Conceito “internacionalização” no Amadeus .....	62
5.1.8.	Conceito “serviços” no Amadeus .....	62
5.1.9.	Conceito “grupos” no Amadeus.....	62
5.2.	Moodle.....	62
5.2.1.	Conceito “atores” no Moodle .....	65
5.2.2.	Conceito “interface” no Moodle .....	66
5.2.3.	Conceito “cursos” no Moodle.....	66
5.2.4.	Conceito “percepção do ambiente” no Moodle .....	67
5.2.5.	Conceito “customização do ambiente” no Moodle.....	68
5.2.6.	Conceito “portabilidade para dispositivos” no Moodle .....	68
5.2.7.	Componente "internacionalização” no Moodle .....	68
5.2.8.	Conceito “serviços” no Moodle .....	69
5.2.9.	Conceito “grupos” no Moodle .....	69
5.3.	Claroline .....	70
5.3.1.	Conceito “atores” no Claroline .....	72
5.3.2.	Conceito “interface” no Claroline.....	73
5.3.3.	Conceito “cursos” no Claroline .....	73
5.3.4.	Conceito “percepção do ambiente” no Claroline.....	74
5.3.5.	Conceito “customização do ambiente” no Claroline .....	74
5.3.6.	Conceito “portabilidade para dispositivos” no Claroline.....	74
5.3.7.	Conceito “internacionalização” no Claroline.....	75
5.3.8.	Conceito “serviços” no Claroline.....	75
5.3.9.	Conceito “grupos” no Claroline.....	75
5.4.	Eureka.....	76
5.4.1.	Conceito “atores” no Eureka.....	78
5.4.2.	Conceito “interface” no Eureka .....	78
5.4.3.	Conceito “cursos” no Eureka .....	79
5.4.4.	Conceito “percepção do ambiente” no Eureka .....	80
5.4.5.	Conceito “customização do ambiente” no Eureka.....	80
5.4.6.	Conceito “portabilidade para dispositivos” no Eureka .....	81
5.4.7.	Conceito "internacionalização” no Eureka .....	81
5.4.8.	Conceito “serviços” no Eureka .....	81
5.4.9.	Conceito “grupos” no Eureka .....	82

5.5.	Sakai .....	82
5.5.1.	Conceito “atores” no Sakai .....	84
5.5.2.	Conceito “interface” no Sakai.....	84
5.5.3.	Conceito “cursos” no Sakai .....	85
5.5.4.	Conceito “percepção do ambiente” no Sakai.....	85
5.5.5.	Conceito “customização do ambiente” no Sakai .....	85
5.5.6.	O Conceito “portabilidade para dispositivos” no Sakai.....	86
5.5.7.	Conceito “internacionalização” no Sakai.....	86
5.5.8.	Conceito “serviços” no Sakai .....	86
5.5.9.	Conceito “grupos” no Sakai.....	87
5.6.	Tidia-ae.....	87
5.6.1.	Conceito “atores” no Tidia-ae.....	90
5.6.2.	Conceito “interface” no Tidia-ae .....	90
5.6.3.	Conceito “cursos” no Tidia-ae .....	91
5.6.4.	Conceito “percepção do ambiente” no Tidia-ae .....	91
5.6.5.	Conceito “customização do ambiente” no Tidia-ae.....	91
5.6.6.	Conceito “portabilidade para dispositivos” no Tidia-ae .....	92
5.6.7.	Conceito “internacionalização” no Tidia-ae .....	92
5.6.8.	Conceito “serviços” no Tidia-ae .....	92
5.6.9.	Conceito “grupos” no Tidia-ae .....	93
5.7.	Análise comparativa entre ambientes .....	93
5.8.	Síntese.....	96
6.	ARQUITETURA DE REFERÊNCIA PARA AMBIENTES VIRTUAIS DE APRENDIZAGEM.....	96
6.1.	Requisitos de Qualidade da Solução Proposta .....	97
6.1.1.	Requisitos de Qualidade como Prioridade Alta .....	97
6.1.2.	Requisitos de Qualidade como Prioridade Média.....	98
6.1.3.	Requisitos de Qualidade como Prioridade Baixa .....	100
6.2.	Arquitetura de Referência da Solução Proposta .....	101
6.3.	Detalhamento da Arquitetura Proposta em UML <i>Components</i> .....	108
6.4.	Sistema de Recomendação Proposto .....	115
7.	REFATORAÇÃO DA ARQUITETURA DO AVA SAKAI UTILIZANDO A ABORDAGEM PROPOSTA.....	119
7.1.	Execução do Sistema de Recomendação.....	121
7.2.	Arquitetura do AVA <i>Sakai</i> gerada pelo sistema de recomendação.....	121
7.2.1.	Detalhamento da arquitetura .....	122

8. CONCLUSÃO E PERSPECTIVAS FUTURAS .....	124
---	-----

# 1. INTRODUÇÃO

Neste capítulo introdutório, apresenta-se uma visão geral da investigação, começando com uma descrição do contexto e sua evolução para a indicação do foco da pesquisa, enfatizando uma problemática, seguindo-se com uma discussão sobre os motivos para este trabalho. Prossegue-se então com a descrição dos propósitos do trabalho e esclarecimento sobre as principais etapas do caminho metodológico adotado. Finalmente, discute-se a relevância da pesquisa, encerrando-se o capítulo com uma síntese do que consta nos capítulos seguintes.

## 1.1. Contexto, Problemática e Motivação da Pesquisa

A presente dissertação se situa na linha de pesquisa modelos computacionais em educação, do Programa Interdisciplinar de Pós-Graduação em Modelagem Computacional de Conhecimento da Universidade Federal de Alagoas. Esta dissertação enfoca a área de Ambientes Virtuais de Aprendizagem (AVA) e faz parte de um projeto mais amplo, denominado AVAL, em execução no Instituto de Computação da Universidade Federal de Alagoas (UFAL). Este trabalho de pesquisa foi aprovado pelo Edital CAPES nº 15 (23/03/2010)<sup>1</sup> – Fomento ao Uso das Tecnologias de Informação nos Cursos de Graduação, cujo título é: Estudo comparativo, proposta, implementação e implantação de uma solução para AVA em Educação Presencial, tendo os orientadores deste trabalho, professor Evandro Costa e o professor Patrick Henrique da Silva Brito como membros da equipe, sendo a coordenação do professor Evandro Costa. Tal projeto está relacionado a um estudo sobre AVA, com foco no seu uso em educação presencial, analisando-os tanto na perspectiva do professor autor quanto do engenheiro de software e desenvolvedor de tais ambientes.

Esta dissertação busca atender aos propósitos deste projeto e, desse modo, objetiva-se uma proposta preliminar que ofereça facilidades aos dois tipos de atores, professor autor e engenheiro de software, tanto em suas atividades de análise, tendo um arcabouço conceitual para compará-los, quanto de construção ou mesmo aprimoramento desses ambientes, dispondo de mecanismos para intervir na definição arquitetural dos ambientes e daí poder evoluir em suas implementações.

---

<sup>1</sup> Este edital busca incentivar a integração e a convergência entre as modalidades de educação presencial e a distância nas Instituições Públicas de Ensino Superior (IES), federais e estaduais, integrantes do Sistema UAB, por meio do estímulo ao uso de tecnologias de comunicação e informação no universo educacional dos cursos de graduação presenciais.

Várias plataformas de AVA já foram desenvolvidas e disponibilizadas, tendo com isso atraído uma quantidade significativa e diversificada de usuários. Partes desses usuários são educadores já experientes e que possuem um entendimento bem refinado e crítico sobre a efetividade de um AVA, ou mesmo tem expectativas sobre quais requisitos um determinado AVA deveria satisfazer, principalmente considerando os avanços tecnológicos, possibilitando a viabilização de diferentes propostas pedagógicas.

A seguir alguns cenários que ilustram lacunas importantes no panorama de AVA existente, levando-se a insatisfações de seus usuários, tanto com alguns dos recursos disponíveis quanto com os que são interessantes, mas ainda não estão oferecidos.

**Cenário 1:** Supondo que um professor deseja utilizar um AVA em sua disciplina de graduação a distância. Esse mesmo professor tem apenas uma noção geral sobre AVA, mas tem uma clareza do que poderia requisitar dele. Assim, um possível caminho a ser seguido pelo professor para atingir esse objetivo, seria:

Passo1- esclarecer-se mais sobre o que é um AVA e o que ele poderia oferecer;

Passo2- fazer um levantamento das funcionalidades e outros requisitos;

Passo3- utilizar algum guia para saber dos recursos e potencialidades de um AVA (por exemplo, acessando um modelo de referência); e

Passo4 - comparar e avaliar os AVA existentes e que estão dentro de suas possibilidades e restrições: se algum AVA, existente, atender à sua necessidade, ele o utilizará, senão, ele precisará elaborá-lo.

**Cenário 2:** Supondo que um professor seja um usuário um pouco experiente no uso de um AVA específico, mas não está satisfeito com o que este AVA lhe oferece. Considerando ainda, que ele esteja então interessado em conhecer outros AVA e compará-los. Assim, para atingir esse objetivo, o professor poderia seguir os passos 2, 3 e 4, apresentados anteriormente.

**Cenário 3:** Neste terceiro cenário, suponhamos que um professor seja usuário, isoladamente, de dois ou mais AVA, ou ainda de um AVA e mais alguns sistemas de informação, como por exemplo, o sistema acadêmico e jogos educacionais. Considerando que ele deseja interoperar os AVA e o sistemas de informação, é necessário saber se os AVA possuem suporte à interoperabilidade. Se não possuírem, poderia ser necessário desenvolver um novo AVA.

**Cenário 4:** O quarto cenário seria supor que um professor conheça os principais AVA e julga que nenhum deles é adequado ao que ele pretende realizar. Considerando nesse caso que ele está disposto a participar de uma equipe para desenvolver um AVA que cumpra o que pretende.

Então, caso seja preciso elaborar e desenvolver um novo, algumas atividades importantes para atingir esse objetivo seriam: fazer o levantamento dos requisitos funcionais; fazer o levantamento dos requisitos não-funcionais; utilizar alguma arquitetura de AVA como referência; e aproximar as escolhas feitas pelo educador, através do guia, a uma arquitetura para o desenvolvedor;

Diante desses cenários apresentados acima, surgem alguns questionamentos: i) Qual seria a representação mais apropriada, em termos de expressividade e legibilidade, de um modelo conceitual de AVA para apoiar o educador na avaliação de uma AVA? ii) Qual seria a representação mais apropriada, em termos de expressividade e legibilidade, de um modelo conceitual de AVA para apoiar o educador na elaboração de uma AVA? iii) Qual seria a representação mais apropriada de uma arquitetura de AVA para apoiar o desenvolvedor de software? Iv) Como aproximar a representação do educador da representação do desenvolvedor de software?

Com isso, observa-se que ainda não há facilidades e ferramentas de software apropriadas para orientar uma participação efetiva dos educadores no desenvolvimento destes ambientes. Faz-se, então, necessário organizar e representar apropriadamente o mencionado entendimento e conhecimento sobre AVAs e suas possibilidades, investindo-se, por exemplo, na elaboração de uma arquitetura de referência, baseada nos AVAs, para dar suporte ao processo de desenvolvimento desses ambientes. Nesse sentido, o trabalho proposto em Crespo et al (1998) focalizou tais aspectos, procurando caracterizar em um mapa conceitual uma representação que servia como uma referência para os AVAs existentes naquele momento. Entretanto, apesar do mérito dessa proposta, atualmente ela já não reflete o estágio corrente desses ambientes, nem do arsenal tecnológico do momento, além disso, apresenta apenas um mapa conceitual e não se preocupa com requisitos. Por tanto, faz-se necessário à elaboração de não só um novo mapa, mas uma arquitetura que contemple o atual estado desses ambientes e do momento tecnológico atual.

O foco desta pesquisa concentra-se em abordar os problemas relacionados à falta de elementos para compreensão de recursos e a pouca de flexibilidade dos AVA atuais com respeito à necessidade de customização em diversos aspectos e adaptação de soluções

particulares, inclusive contemplando requisitos pedagógicos, dificuldade para integração com outras ferramentas e dados externos aos AVA. Além disso, o AVA deve oferecer segurança de acesso tanto aos alunos quanto aos professores, deve permitir escalar facilmente o número de usuários do sistema e deve ser acessível a partir de qualquer sistema operacional. Tais requisitos podem ser encarados como desafios a serem alcançados por sistemas educacionais, para atender de fato ao que se espera deles: apoio a educação de qualidade e em larga escala.

Assim, nesta dissertação apresenta-se uma solução para amenizar parte dos problemas existentes, anteriormente mencionados, focalizando na definição de um mapa de referência que conta com um mapa conceitual destinado a auxiliar educadores a escolher serviços de um AVA, que atenda suas necessidades. Ademais, oferece ainda uma arquitetura de referência que se presta a ajudar os desenvolvedores a implementar as demandas dos usuários. Assim, resultando em um arcabouço adequado para realizar avaliações e comparações entre AVA, além de servir para orientar e viabilizar o desenvolvimento de novos ambientes atendendo demandas específicas de educadores.

Observando-se o panorama nacional, constata-se que o Brasil vive um momento de expansão no uso das TIC na modalidade de educação a distância, particularmente no que tem ocorrido no Sistema Universidade Aberta do Brasil (UAB). Além disso, mais recentemente verifica-se também a inserção das TIC nos cursos presenciais, servindo para viabilizar atividades complementares ao que ocorre na sala de aula tradicional. Tudo isso se estende ainda para pós-graduação, existindo portarias que disciplinam essas atividades. Com relação à modalidade presencial, por exemplo, a portaria do Ministério da Educação (MEC) nº 4.059/04, incentiva as instituições de ensino, de forma facultativa, à inclusão de atividades não presenciais nos cursos de graduação, sendo permitida até o limite de 20% da carga horária do curso.

Neste sentido, a concretização efetiva de tais possibilidades requer um investimento significativo nas TIC, o que confere aos AVA, um papel de destaque nessa gama de possibilidades.

AVA são sistemas baseados na web, destinados a apoiar atividades educacionais nas modalidades presencial e a distância. Conforme Schlemmer (2005), o ambiente deve integrar espaços, permitir a construção, a livre exploração e a descoberta do conhecimento, servindo como ponto de encontro onde os agentes se reúnem para desenvolver atividades. “O desenvolvimento dos AVA tem sua origem na segunda metade da década de noventa, quando os primeiros ambientes de Educação Baseada na Web foram desenvolvidos e utilizados em

curso a distância” (SARMENTO, 2011, p.1). Entre os AVA atualmente em uso, destacam-se o Amadeus, Moodle, Eureka, Claroline, Sakai e Tidia-ae.

## **1.2. Objetivos**

Este trabalho tem por objetivo geral propor uma solução de referência para AVA, sendo composta de: i) um mapa conceitual, do estado da arte e da prática desses ambientes, sendo desenvolvido com o objetivo de se conceber uma visão do AVA para o educador; ii) uma arquitetura de referência, baseada em requisitos não-funcionais definidos pelo mapa ISO 9126, que busca traduzir a visão do desenvolvedor; e iii) um sistema de recomendação, que visa aproximar a visão do educador da visão do desenvolvedor, mapeando as escolhas do educador, em o que o desenvolvedor precisa para desenvolvê-lo.

Especificamente, objetiva-se:

- Propor um mapa conceitual de referência para AVA, desenvolvido com base em uma análise de alguns AVA, que seja uma representação mais apropriada, em termos de expressividade e legibilidade, para apoiar o educador na avaliação e na elaboração de uma AVA;
- Validar o mapa conceitual através de uma análise comparativa realizada em cinco AVAs, ou seja, verificando-se o quanto cada ambiente contempla de cada componente do mapa conceitual;
- Estabelecer os requisitos não-funcionais, juntamente com alguns cenários de aplicação destes cenários, que devem ser considerados no desenvolvimento de um AVA;
- Propor uma arquitetura de software baseada em requisitos não-funcionais definidos pelo mapa ISO 9126 e detalhamento da arquitetura utilizando o diagrama de componentes elaborado com base no mapa conceitual;
- Desenvolver um sistema de recomendação que mapeia as escolhas do educador, em relação às funcionalidades do AVA, em o que o desenvolvedor precisa para desenvolvê-lo, visando aproximar a visão do educador com a visão do desenvolvedor.

## **1.3. Relevância**

O mapa proposto deste trabalho contribui com subsídio para o desenvolvimento de AVA personalizados. Assim, o mapa conceitual pode ser utilizado como meio facilitador de

comunicação entre educadores e desenvolvedores de softwares, pois através dos componentes do mapa conceitual (compreensível ao educador) pode-se gerar uma arquitetura de software (compreensível ao desenvolvedor de software).

Como a arquitetura de software gerada reflete os requisitos passados pelo educador através do mapa conceitual, ela é usada no desenvolvimento de AVA que atende a necessidades de disciplinas, instituição ou departamento. Além disso, apesar deste mapa refletir o estado da arte dos AVA atualmente, nada impedirá que estes ambientes evoluam, pois segundo Sarmiento (2011), os AVA não podem ser considerados versões definitivas, pois estão sempre sendo melhorados com a incorporação de novas funcionalidades. Mesmo assim, acredita-se que este mapa possa acompanhar a evolução desses ambientes, pois como o mapa conceitual foi elaborado utilizando componentes e o detalhamento da arquitetura de software foi construído com a utilização de diagrama de componentes, é possível remover ou acrescentar componentes.

Assim, para viabilizar a transformação da visão do educador (o mapa conceitual) na visão do desenvolvedor (a arquitetura de referência e seus detalhamentos) foi desenvolvido um sistema de recomendação. Este sistema mapeia cada escolha do usuário educador, uma arquitetura de software com seus detalhamentos, para o desenvolvedor.

#### **1.4. Aspectos Metodológicos**

Esta dissertação segue a partir de uma pesquisa onde realizou-se uma investigação em direção à conceber os objetivos gerais propostos para este trabalho deu-se da seguinte maneira. Seguiu uma metodologia que em relação ao quanto aos objetivos específicos é classificada como foi realizada uma pesquisa exploratória. Este tipo de pesquisa tem “como objetivo proporcionar maior familiaridade com o problema” (GIL, 1991, p. 45). Para Ponte (2006) a pesquisa exploratória permite uma maior familiaridade com o problema, com o objetivo de torná-lo mais explícito ou a facilitar a construção de hipóteses ( Theodorson e Theodorson, 1970) (Babbie, 1986). Além disso, ela pode utiliza fontes que darão base ao assunto abordado, como é o caso da pesquisa bibliográfica e das entrevistas com pessoas que tiveram experiências práticas com o problema pesquisado.

Segundo Ponte (2006, p.5) o delineamento da pesquisa corresponde ao seu planejamento numa dimensão mais ampla; ou seja, nesse momento o investigador estabelece os meios técnicos da investigação. Assim, quanto ao delineamento, a pesquisa classificou-se em: i) pesquisa bibliográfica, com os dados obtidos de livros, revistas científicas, teses,

relatórios científicos, cuja autoria é conhecida, sobre diversos AVA; ii) pesquisa experimental dos seguintes ambientes: Amadeus, Moodle, Eureka, Claroline, Sakai e Tidia-ae.

Quanto à natureza, a pesquisa classificou-se em qualitativa (PONTE, 2006, p.5). Uma das fases mais importantes da pesquisa é a coleta de dados ocorre após a revisão bibliográfica. Nessa fase, para este trabalho, foram empregadas as seguintes técnicas: entrevista e observação.

As entrevistas realizadas para este trabalho foram do tipo semidirigida (Quivy e Campenhoudt, 1995), que é orientada por perguntas-guias, relativamente abertas, são colocadas pela ordem que a conversa, entre ambos, encaminhar. Essas entrevistas foram realizadas com alunos e professores usuários de AVA e esses usuários possuíam diversos perfis.

Seguindo a metodologia apresentada anteriormente, conseguiu-se extrair dados qualitativos. Esses dados após processados originaram os requisitos funcionais para subsidiar a construção de um mapa conceitual, para representar a diversidade de recursos dos AVA existentes. A partir da identificação e análise dos requisitos funcionais, especificou-se um mapa conceitual contemplando as variabilidades de características dos AVA avaliados, além das necessidades dos atores entrevistados.

Com respeito à busca pela definição de uma arquitetura de referência, inicialmente foram especificados os requisitos de qualidade (não-funcionais) para os AVA com base nas diretrizes propostas na norma ISO 9126/IEC, de onde se extraem diretrizes de qualidade que podem ser ajustadas de acordo com a necessidade do usuário. A partir desses requisitos, a arquitetura de referência pode ser especificada baseada em refinamentos de acordo com as necessidades não-funcionais do usuários, como por exemplo número de usuários simultâneos(escalabilidade).

Após as etapas supracitadas, veio o trabalho de elicitación de conhecimento a fim construir uma base para servir de apoio ao sistema de recomendação baseado em conhecimento. Finalmente, para avaliar a solução proposta, via mapa conceitual e arquitetura, foi conduzido um estudo de caso instanciando o mapa conceitual e a arquitetura de referência no contexto de um AVA específico, atendendo os requisitos estabelecidos pelo ator responsável pela proposição de um curso. Além disso, foi considerado um cenário no qual

esse ator necessitou de incluir novos requisitos funcionais e de qualidade (não-funcionais) (facilidade de evolução).

## **1.5. Estrutura da Dissertação**

A estrutura da dissertação aqui apresentada é a que se segue:

- Capítulo 2: Fundamentação Teórica apresenta conceitos sobre engenharia de software e sistemas baseados em conhecimento.
- Capítulo 3: Trabalhos Relacionados apresenta alguns trabalhos relacionados à proposta de trabalho, sendo subdividido em trabalho relacionado ao mapa conceitual e trabalho relacionado à arquitetura;
- Capítulo 4: Mapa conceitual apresenta o mapa conceitual subdividido em três: mapa conceitual geral, mapa conceitual para sob a visão do ator professor e mapa conceitual para sob a visão do ator aluno.
- Capítulo 5: Avaliação comparativa realizadas em seis ambientes. Assim, será apresentada uma versão do mapa conceitual, para cada AVA. Ao final será apresentada uma tabela comparando os ambientes entre si, com relação a algumas principais características extraídas do mapa conceitual.
- Capítulo 6: Arquitetura de Referência para AVA, cuja elaboração foi baseada no mapa conceitual e em requisitos não-funcionais, o detalhamento da arquitetura e finalmente, o sistema de recomendação.
- Capítulo 7: Refatoração da Arquitetura do AVA Sakai Utilizando a Abordagem Proposta, apresenta um cenário de uso da solução proposta.
- Capítulo 8: Conclusão e Trabalhos Futuros, no qual os resultados, conclusões e trabalhos futuros são apresentados.

## **2. FUNDAMENTAÇÃO TEÓRICA**

Apresenta-se, a seguir conceitos fundamentais envolvidos no mesmo. Com esse propósito, neste capítulo são apresentados alguns conceitos de Arquitetura de Software com Estilos e Padrões Arquiteturais e Sistemas Baseados em Conhecimento.

### **2.1. Aspectos de Engenharia de Software**

Uma arquitetura de software define o sistema em termos de seus componentes arquiteturais, que representam unidades abstratas do sistema, criando um nível abstração; a comunicação e interação entre os componentes se materializa através dos conectores, com os atributos e funcionalidades de cada um (SOMMERVILLE, 2006). Como os conectores conhecem o fluxo interativo entre os componentes do sistema, podem estabelecer protocolos de comunicação e coordenar a execução das atividades que envolvam mais de um componente do sistema.

A visão estrutural do sistema em um alto nível de abstração proporciona benefícios importantes, que são indispensáveis para o desenvolvimento de sistemas de software complexos. Os principais benefícios são:

- i) A organização do sistema como uma composição de componentes lógicos;
- ii) A antecipação da definição das estruturas de controle globais;
- iii) A definição da forma de comunicação e composição dos elementos do projeto;  
e
- iv) O auxílio na definição das funcionalidades de cada componente projetado.

Além disso, uma propriedade arquitetural representa uma decisão de projetor que foi tomada para atender a algum requisito não-funcional do sistema, que quantifica determinados aspectos do seu comportamento, como confiabilidade, reusabilidade e modificabilidade (BASS, CLEMENTS, KAZMAN, 2003) (SOMMERVILLE, 2006).

A presença de uma determinada propriedade arquitetural pode ser obtida através da utilização de estilos arquiteturais que possam garantir a preservação dessa propriedade durante o desenvolvimento do sistema (SHAW e GARLAN, 1996) (MONROE et al, 1997). Um estilo arquitetural caracteriza uma família de sistemas que são relacionados pelo

compartilhamento de suas propriedades estruturais e semânticas. Esses estilos definem inclusive restrições de comunicação entre os componentes do sistema.

As propriedades arquiteturais são derivadas dos requisitos do sistema e influenciam, direcionam e restringem todas as fases do seu ciclo de vida. Sendo assim, a arquitetura de software é um artefato essencial no processo de desenvolvimento de softwares modernos, sendo útil em todas as suas fases (CHEN, WANG, MEI e YANG, 2002). A importância da arquitetura fica ainda mais clara no contexto do desenvolvimento baseados em componentes. Isso acontece, uma vez que na composição de sistemas, os componentes precisam interagir entre si para oferecer as funcionalidades desejadas. Além disso, devido à diferença de abstração entre a arquitetura e a implementação de um sistema, um processo de desenvolvimento baseado em componentes deve apresentar uma distinção entre os conceitos de componente arquitetural, que é abstrato e não é necessariamente instanciável; e componente de implementação, que representa a materialização de uma especificação em alguma tecnologia específica e deve necessariamente ser instanciável.

## **2.2. Sistemas Baseados em Conhecimento**

Sistemas Baseados em Conhecimento são programas de computador que usam o conhecimento representado explicitamente para resolver problemas (REZENDE, 2003, p. 15). Estes sistemas possuem uma base de conhecimento separada do código, assim qualquer alteração na regra de negócio pode ser facilmente adicionada à base.

Para Burke (2002 apud OLIVEIRA, 200-?) um sistema de Recomendação é um sistema que recomende individualmente, ou que guie o usuário de forma a apresentar algo de seu interesse dentre uma grande quantidade de opções.

Recomendação baseada em conhecimento (BURKE, 2007), tem como principal característica uma base de conhecimento sobre o domínio de aplicação. Assim, esse tipo de sistema é formado, basicamente, por três partes:

- Interface: o usuário interage com o sistema através desta interface que simplifica a comunicação e o abstrai da complexidade do sistema;
- Base de conhecimento: contém o conhecimento de um domínio particular de aplicação. Sendo este conhecimento representado, geralmente, na forma de regras *se...então...* ;

- Motor de inferência: aplica o conhecimento à solução de problemas reais. Assim, uma vez construída a base de conhecimento, o motor de inferência extrai informações da base de conhecimento de acordo com o que é passado pelo usuário através da interface.

### 3. TRABALHOS RELACIONADOS

Neste capítulo serão apresentadas algumas propostas para ambientes virtuais de aprendizagem que possuem relação com o mapa proposto por este trabalho. Sendo expostos alguns trabalhos suportes, ou seja, que serve como base para este trabalho e outros são considerados uma tendência para o mapa apresentado nesta dissertação.

#### 3.1. Perspectivas de um modelo conceitual

Este trabalho tem como título: Um Mapa Conceitual Compatível com a Plataforma EDUCOM/IMS para Comparação de Ambientes de Educação na WEB e apresenta um mapa, formado por nove componentes, ilustrado na Figura 1. Desta forma a existência de cada um desses componentes em um ambiente significa que tal ambiente implementa a funcionalidade especificada pelo componente. (Crespo *et. al*, 1998, p. 3).

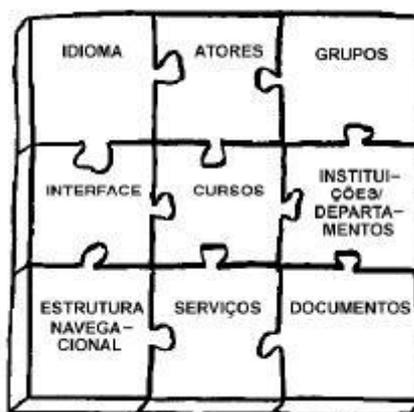


Figura 1 - Modelo conceitual baseado em componentes (CRESPO et al, 1998, p. 3)

O componente “cursos” que verifica se o ambiente permite a criação e manutenção de cursos, “atores” que são as pessoas que interagem com o ambiente, “serviços” que verifica se o ambiente provém à funcionalidade necessária para o curso, “documentos” são os artefatos manipulados pelos serviços. Já o componente “grupos” é a possibilidade de se definir grupos, possibilitando trabalho cooperativo (CRESPO et al, 1998, p. 3).

O componente “instituições e departamentos” é capacidade de definir e customizar o ambiente para diversas instituições e departamentos, “idiomas” suporte a autoria e consumo de cursos em vários idiomas, “interface” que é a capacidade de customização da interface do ambiente e “estrutura navegacional” que é a capacidade de customização da estrutura navegacional do ambiente. (CRESPO et al, 1998, p. 3)

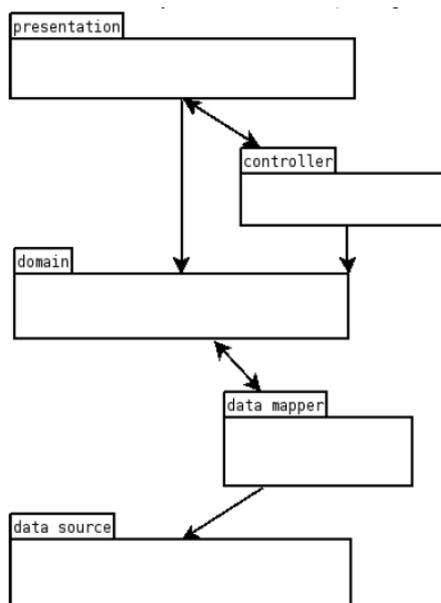
Ao contrário desta dissertação, neste trabalho não são tratados refinamentos para os componentes, ou seja, os subcomponentes. Assim, alguns componentes deste trabalho, também estão nesta dissertação, sendo que nesta última, os componentes possuem vários subcomponentes, como é o caso de “cursos”, “grupos”, “serviços” e “atores”. Já os componentes “idiomas”, “interface”, “estrutura navegacional”, “documentos” e “instituições e departamentos”, nesta dissertação, estão representados em outros componentes.

Este trabalho também faz uma análise comparativa entre o modelo e os ambientes virtuais de aprendizagem WCB, Web-CT, LearningSpace, Virtual-U, LiveBOOKS e Aulanet. Ao final, são mostrados alguns detalhes de implementação.

### **3.2. Perspectivas de uma Arquitetura**

Este trabalho intitulado *Agile and Open E-Learning Systems Architecture*, propõe uma arquitetura de software para AVAS. Assim, ele analisa as características de desenvolvimento de sistemas para *e-learning* e examina as vantagens da utilização dos princípios de desenvolvimento ágil (SWEENEY, 2008).

A arquitetura foi projetada em camadas lógicas e como podemos observar na Figura 2 foi modelada utilizando diagrama de pacotes.



**Figura 2 – Arquitetura em camadas lógicas (SWEENEY, 2008, p. 62)**

No entanto, diferente desta dissertação esse trabalho não considera requisitos não-funcionais. Por essa razão, essa arquitetura foi projetada utilizando, basicamente, o padrão MVC (Buschman *et al.*, 1996), enquanto que esta dissertação, por considerar requisitos não-funcionais utiliza padrões que atendem cada requisito não funcionais por ordem de prioridade.

Nesta dissertação o detalhamento interno dos componentes arquiteturais, foi feito de maneira sistemática, utilizando o processo UML *Components*, com base no mapa conceitual. Já esse trabalho relacionado não deixou claro como os "sub-componentes" foram identificados. Além do mais, esta dissertação, nos detalhamentos, continuará seguindo a mesma abstração de componentes/módulos, enquanto no trabalho relacionado está mais próximo de classe OO.

Contudo, vale ressaltar que uma das contribuições mais importantes deste trabalho é o envolvimento efetivo do educador no processo de desenvolvimento do AVA, enquanto que neste trabalho relacionado não houve essa preocupação.

## 4. O MAPA CONCEITUAL PARA AVAS

Neste trabalho são apresentados três mapas conceituais de um AVA: (i) o primeiro é considerado o mapa geral, sendo independente da visão do ator ou usuário; (ii) o segundo é um mapa derivado do geral, mas que está relacionado com a visão do ator “professor”; (iii) e outro também derivado do geral, mas agora com relacionado à visão do ator “aluno”. Foram escolhidos estes dois atores para ser construído um mapa conceitual, porque para Kenski (2005) as pessoas envolvidas no processo educativo são os professores e os alunos.

### 4.1. Metodologia

A concepção do mapa conceitual geral proposto seguiu uma metodologia que em relação ao quanto aos objetivos específicos é classificada como foi realizada uma pesquisa exploratória. Este tipo de pesquisa tem “como objetivo proporcionar maior familiaridade com o problema” (GIL, 1991, p. 45). Para Ponte a pesquisa exploratória permite uma maior familiaridade com o problema, com o objetivo de torná-lo mais explícito ou a facilitar a construção de hipóteses (Ponte).

Esse tipo de pesquisa tem como principal objetivo o aprimoramento de idias ou a descoberta de intuições, novas ideias.

Além disso, ela pode utiliza fontes que darão base ao assunto abordado, como é o caso da pesquisa bibliográfica e das entrevistas com pessoas que tiveram experiências práticas com o problema pesquisado.

- i) Análise da literatura através do levantamento da documentação, teses, dissertações e de artigos dos seguintes ambientes: Amadeus, Moodle, Eureka, Claroline, *Sakai* e Tidia-ae. Sendo obtido como resultado a primeira versão do mapa conceitual e os primeiros requisitos não-funcionais. (autores utilizados)
- ii) Na segunda etapa foram realizadas algumas entrevistas, tanto com profissionais, quanto com professores das áreas ciência da computação e educação. Assim, através da aplicação de questionário e com a apresentação da primeira versão do mapa, a cada entrevistado, obteve-se a segunda versão do mapa e um aprimoramento nos requisitos não-funcionais.
- iii) Na terceira e última etapa, foi realizada uma análise exploratória como os mesmos ambientes citados anteriormente, o Moodle, Amadeus, Eureka, Claroline, *Sakai* e Tidia-ae.

Optou-se por esses ambientes, primeiramente, porque estão em uso e disponíveis para testar e segundo, porque são ambientes de código aberto. Esta análise procedeu-se da seguinte forma:

- i) Foi visitado cada ambiente com a visão dos diferentes atores;
- ii) Foram testados e utilizados cada serviço e recurso disponibilizado no ambiente para cada ator;
- iii) Foram extraídas as características principais de cada ambiente. Por fim, como resultado obteve-se a última versão do mapa apresentada nesta dissertação.

A partir do mapa conceitual, entrevistas e todas as pesquisas apresentadas anteriormente atingiu-se a arquitetura apresentada no Capítulo 6.

Nas próximas seções serão discutidos todos os refinamentos para os componentes que constituem o mapa.

## **4.2. O Mapa Conceitual Geral**

O mapa geral é apresentado na Figura 3, ele é formado por nove conceitos, que idealmente deveriam ser considerados no desenvolvimento de um ambiente virtual de aprendizagem. Cada conceito pode ser decomposto em subconceitos.

Nas Figuras 4 e 5 é proposto o mapa conceitual completo, ou seja, como todos os subconceitos, foi dividido em duas partes para ficar mais legível. Além disso, cada conceito possui no máximo cinco níveis de subconceitos, por exemplo, o conceito “atores” possui dois níveis, no primeiro nível está o conceito e no segundo nível estão os subconceitos “fixo” e “flexível”. O mapa apresentado nesta seção é considerado o mapa conceitual geral para um AVA, sendo assim, independente da visão do ator ou usuário.

A hierarquia de características apresentada no mapa conceitual das Figuras 4 e 5 possuem dois papéis importantes. O primeiro deles refere-se à organização do mapa em si, através da definição de conceito e subconceito. O segundo papel refere-se à interdependência entre conceito e subconceito, semelhante ao conceito de interação entre características (tradução livre do inglês *feature interaction*). Para ilustrar esse segundo papel, a hierarquia apresentada denota, por exemplo, que o subconceito "controle de visibilidade" só pode ser selecionado caso o conceito "grupo" também tenha sido selecionado. Além das dependências inerentes à hierarquia, a notação definida no mapa conceitual proposto permite a definição de outras dependências envolvendo conceitos de hierarquias diferentes (linha tracejada). Por exemplo, além de depender do conceito "grupo", o subconceito "interação intergrupo" também depende do subconceito "serviços de comunicação".

Figura 3 Mapa Conceitual do Ambiente

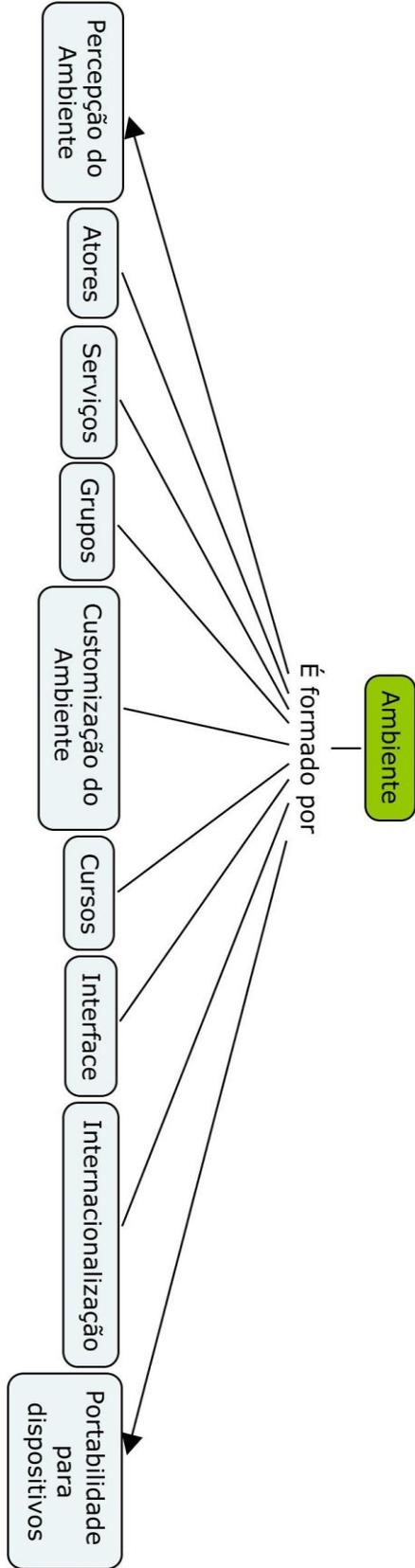
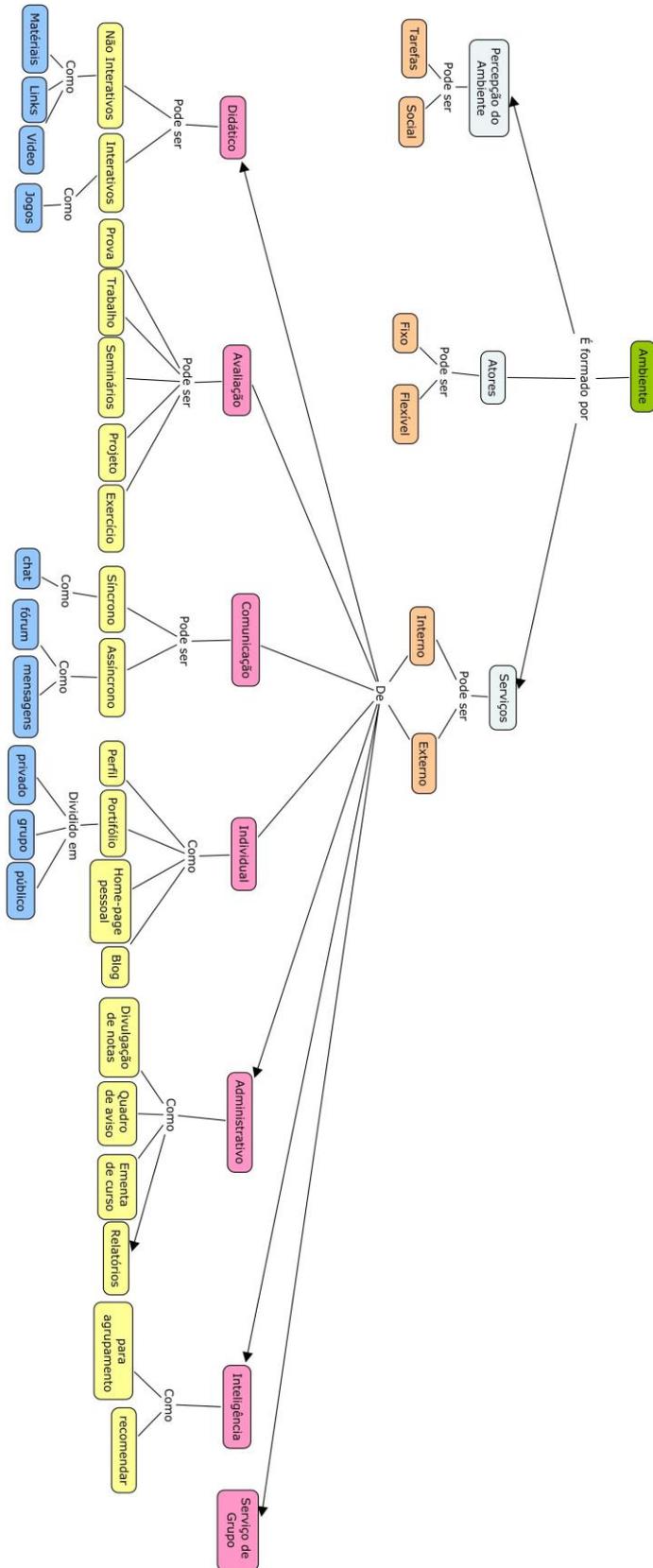




Figura 5 - Mapa Conceitual Geral do Ambiente Completo – Parte 2

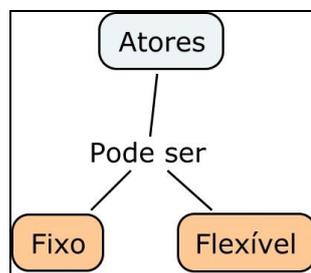


#### 4.2.1. Conceito “atores”

Este conceito, representado pela Figura 6, é considerado a existência de atores que interagem com o ambiente, como, por exemplo, professor, estudante e tutor. Esses atores podem existir de duas formas:

- a) **Fixa**, quando os atores já existem a priori no ambiente e não podem ser alterados, removido ou adicionado;
- b) **Flexível**, quando o ambiente permite a adição, remoção e modificação de atores.

Figura 6 – O Conceito Atores



#### 4.2.2. Conceito “interface”

O conceito “interface”, representado pela Figura 7, é responsável por avaliar a interface do AVA. Este conceito pode ser dividido em subconceitos:

- a) A **usabilidade** considera todos os aspectos ergonômicos da área interface homem máquina utilizados no ambiente. Para Moreira (2011, p.5) pode ser entendido como um requisito de qualidade que representa a capacidade da ferramenta, sistema ou software, de ser entendível, de ser utilizável e atrativo para o usuário, quando usado sob condições específicas.

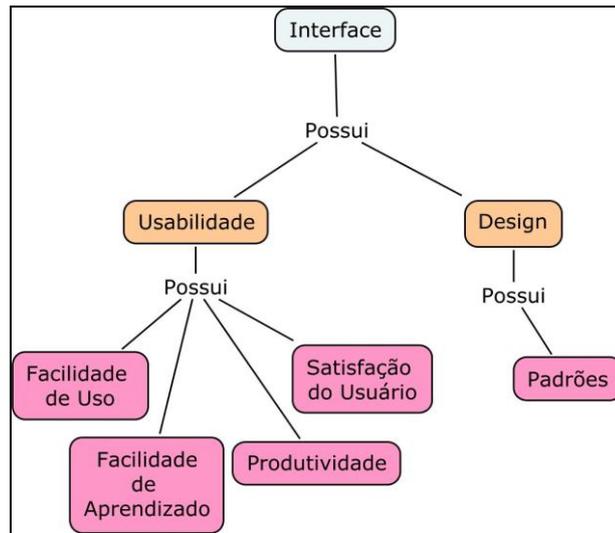
Segundo Lima (2007, citado por Ferreira, 2007) a usabilidade de um sistema é um conceito que se refere à qualidade da interação de sistemas com os usuários e depende de vários aspectos. Aspectos como:

- **Facilidade de aprendizado do sistema:** o tempo e o esforço do usuário para atingir um bom nível de desempenho.
- **Facilidade de uso:** analisa o esforço físico e cognitivo do usuário na interação com o ambiente, mede-se a velocidade e a quantidade de erros cometidos durante a execução de uma atividade.
- **Satisfação do usuário:** avalia o grau de satisfação do usuário ao trabalhar com o ambiente.

- **Produtividade:** averigua se o usuário consegue ser mais produtivo utilizando ambiente.

b) O **Design** trata da utilização de cores, ícones e símbolos utilizados pelo ambiente que possam de alguma forma, maximizar o aprendizado.

**Figura 7 - O Conceito Interface**



#### 4.2.3. Conceito “cursos”

Este componente, representado na Figura 8, busca analisar o suporte que o ambiente oferece para a estrutura e organização de cursos. A organização e estruturação dos cursos podem ocorrer das seguintes formas:

a) **Padrão**, quando ela é fixa para todo e qualquer curso. Como, por exemplo, determinar que, independente do curso, todos eles deverão conter os serviços fórum, chat e tarefas.

b) **Flexível**, quando na estrutura do curso existe a flexibilidade tanto de escolher os serviços que serão utilizados no curso, como também o fluxo das atividades.

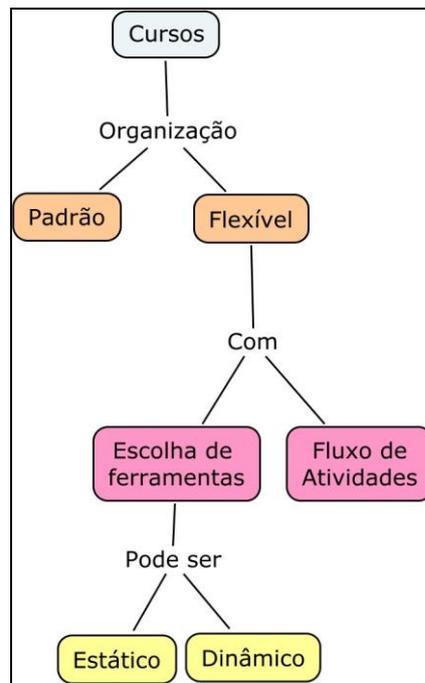
i) Na **escolha de serviços** o professor tem autonomia para determinar, por exemplo, se vai utilizar um fórum ou chat no curso, essa escolha pode ser feita de duas formas:

- **Estática:** cada professor, no início do curso, pode personalizá-lo escolhendo quais serviços serão utilizados, porém não poderá realizar alterações após o início do mesmo.
- **Dinâmico:** o professor poderá realizar modificações, ou seja, adicionar e remover serviços, a qualquer momento do curso.

ii) **Fluxo de atividades:** Cada professor pode escolher a sequência das ferramentas que serão trabalhadas.

- Escolher a data de início e término das atividades;
- Determinar a sequência das atividades e se a mesma deve ser flexível. Assim, professor pode alterar a ordem, por exemplo, colocando uma atividade que, inicialmente, seria realizada no início do curso, para ser realizada no final.

**Figura 8 – O Conceito “Cursos”**

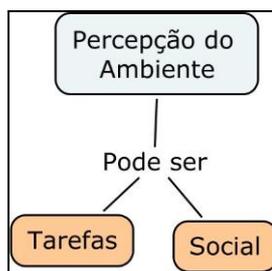


#### 4.2.4. Conceito “percepção do ambiente”

Neste conceito, representado pela Figura 9, são tratados aspectos de percepção de tarefas e atividades, individuais e do grupo, como também a percepção social dentro do ambiente. Para Pinheiro (2001, citado por Silva, 2010) “percepção refere-se a ter conhecimento, ter ciência das atividades do grupo, das atividades que influenciarão o trabalho como um todo”. Assim, temos:

- Percepção Social:** busca identificar e expor para o usuário informações dos demais membros, como, por exemplo, indicar se existe alguém online ou se o professor está online no momento;
- Percepção de Tarefas:** este subconceito analisa se o ambiente mostra ao usuário as atividades e tarefas que o mesmo tem pendente, ou seja, que ainda não foram respondidas ou enviadas. Elas podem ser individuais, aquelas atividades e tarefas que devem ser realizadas individualmente, ou em grupo, aquelas atividades e tarefas que devem ser realizadas pelo grupo do qual o usuário participa;

**Figura 9 - O Conceito “Percepção do Ambiente”**



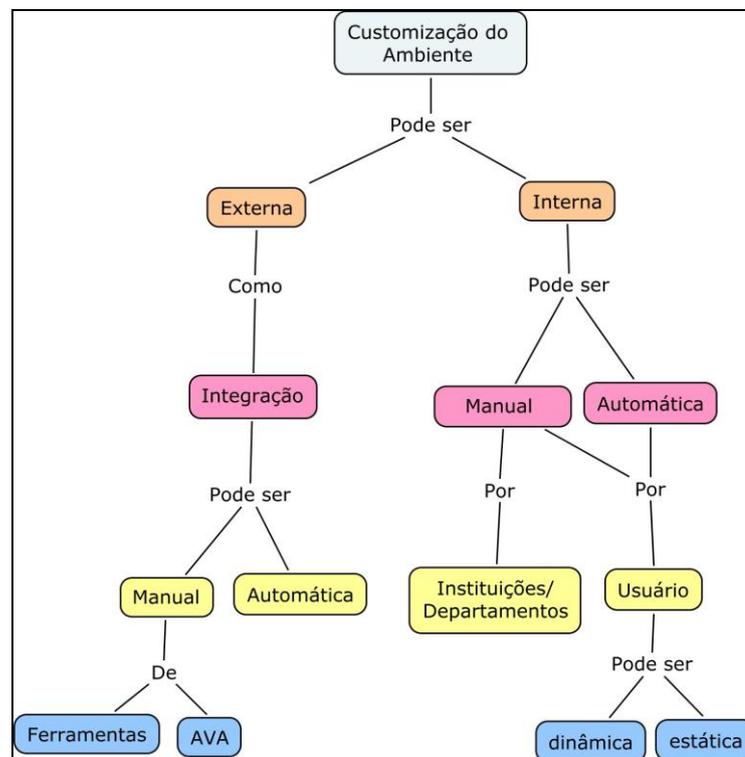
#### 4.2.5. Conceito “customização do ambiente”

Para Campos (2008) um ambiente virtual de aprendizagem é um local onde deve ocorrer troca de informação de uma forma fácil e adaptável. Sendo assim este conceito, representado pela Figura 10, abrange todas as mudanças que podem ocorrer na organização, conteúdo e interface do ambiente. Essa customização pode ser especificada como interna ou externa. Sendo assim, temos:

- a) **Externa:** são customizações que envolvem a integração do ambiente com outros sistemas.
- i) **Integração:** acoplar uma ferramenta ou sistema ao AVA. Pode ser feita:
- **Manual:** Com interferência do usuário ou do desenvolvedor (em nível de código);
    - **Ferramenta:** Integração de ferramentas;
    - **AVA:** Integração com outro AVA, ou seja, interoperabilidade, pois para Silva e Barreto (2008) esse é uma característica importante, já que os cursos à distância tem produzido uma extensa gama de conteúdo na rede. Entretanto, tal conteúdo é geralmente imbricado nos AVAs atuais. Não havendo assim, compartilhamento de conteúdos.
  - **Automática:** Integração automática feita pelo AVA. Exemplo: Serviços Semânticos.
- b) **Interna:** são todas aquelas customizações unicamente relacionadas com o ambiente, podem ser classificadas como:
- i) **Manual:** adaptações feitas no ambiente com interferência de outrem. Podem ser feitas para:
- **Usuário:** permite ao usuário adaptar o ambiente de acordo com suas preferências;
  - **Estática:** Adaptação do ambiente pelo usuário, ou seja, o usuário poderá mudar a interface do ambiente, no início do curso. Exemplo: mudar um fórum de lugar, mudar um menu de lugar, etc.

- **Dinâmica:** Adaptação do ambiente pelo usuário, ou seja, o usuário poderá mudar a interface do ambiente, a qualquer momento. Exemplo: mudar um fórum de lugar, mudar um menu de lugar, etc.
  - **Instituições e Departamentos:** É a customização do ambiente para atender diferentes instituições, departamentos e cursos.
- ii) **Automática:** o ambiente modifica-se automaticamente adaptando-se de acordo com as características e preferências de outrem.
- **Usuário:** modifica-se de acordo com as características e preferências do usuário. Podem acontecer de forma:
    - **Estática:** Adaptação do ambiente para cada usuário do ambiente. Adaptar interface e conteúdo mostrado, no início do curso.
    - **Dinâmica:** Adaptação do ambiente para cada usuário do ambiente. Adaptar interface e conteúdo mostrado, a qualquer momento.

Figura 10 - O Conceito “Customização do Ambiente”



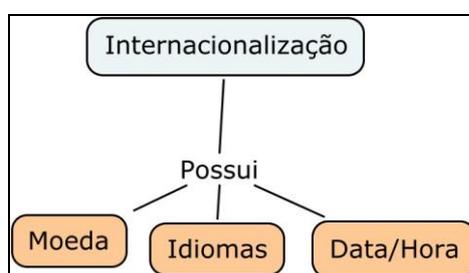
#### 4.2.6. Conceito “portabilidade para dispositivos”

A portabilidade para dispositivos indica se o ambiente oferece suporte para ser acessado a partir de diferentes dispositivos, por exemplo, dispositivos móveis, celulares e TV Digital. Este conceito não possui especificação mais detalhada, mas pode ser observado na Figura 3.

#### 4.2.7. Conceito “internacionalização”

Este conceito, representado pela Figura 11, refere-se à capacidade do ambiente oferecer suporte para que seus cursos possam ser acessados em diferentes idiomas, e possua formato para diferentes moedas e datas.

**Figura 11 - O Conceito “Internacionalização”**



#### 4.2.8. Conceito “serviços”

O conceito “serviços”, representado pelas Figuras 12 e 13, trata dos recursos utilizados no desenvolvimento do curso, podem ser classificados como:

- i) **Interno:** Serviço interno pode ser qualquer serviço usado no ambiente, desde que ele seja nativo do ambiente, ou seja, pode ser serviço didático, administrativo, de grupo e etc., no entanto precisa ter sido desenvolvido junto ao ambiente.
- ii) **Externo:** Serviço externo pode ser qualquer serviço usado no ambiente, desde que ele seja acoplado ao ambiente, ou seja, pode ser serviço didático, administrativo, de grupo e etc., no entanto precisa ter sido desenvolvido fora do ambiente. Um exemplo que merece destaque, nesta categoria de serviço, são as redes sociais. Assim, um ambiente virtual de aprendizagem poderia integrar as redes sociais como um serviço externo de comunicação, onde seriam divulgadas notícias e postados comentários sobre materiais contidos no AVA.

Sendo externo ou interno, os serviços ainda podem ser classificados quando à sua finalidade, ou seja, quanto à intensão e objetivo da utilização do mesmo.

- **Serviços Administrativos:** permitem que determinados atores, por exemplo, professor ou administrador, possam gerenciar o curso. Alguns exemplos de serviços administrativos são: Agenda do curso, Quadro de avisos, Ementa de curso, Divulgação de notas, Controle do progresso do aluno e etc.

- **Serviços Didáticos:** tem como único objetivo viabilizar o aprendizado e podem ser classificados como:
  - **Interativos:** serviços dinâmicos, como por exemplo, jogos.
  - **Não interativos:** serviços estáticos, como por exemplo, materiais, links e vídeos.
- **Serviços de Avaliação:** possuem como finalidade avaliar o desempenho do aluno. Alguns exemplos são provas, autoavaliação, exercícios e tarefas.
- **Serviços de Comunicação:** são todos serviços que, de alguma forma, estabelecem comunicação entre os membros do ambiente, esta comunicação poderá ocorrer de forma:
  - **Síncrona:** os alunos ou professores precisam está conectados ao ambiente ao mesmo tempo. Um exemplo deste tipo de serviço é o chat.
  - **Assíncrona:** os alunos ou professores não precisam está conectados ao ambiente ao mesmo tempo. Um exemplo deste tipo de serviço é o fórum.
- **Serviços Individuais:** são os serviços que cada membro do ambiente utiliza, de modo pessoal, para armazenar suas informações e materiais. São exemplos de serviços individuais: homes-page pessoal, arquivos de imagem indexados, perfis de aluno e professor e portfólio. Sendo que este pode se dividir em:
  - **Portfólio privado:** materiais que somente o usuário tem acesso. No caso do aluno, materiais inacabados aos quais ele não deseja que o professor tenha acesso. No caso do professor, materiais que ele ainda vai passar para os alunos.
  - **Portfólio protegido:** materiais que só o professor e o aluno têm acesso;
  - **Portfólio de grupo:** espaço reservado ao compartilhamento de materiais pelo grupo;
  - **Portfólio público:** materiais que todos dentro do ambiente podem visualizar.
- **Serviços de Grupo:** oferecem suporte a formação e organização de grupos.
- **Serviços de Inteligência:** serviços que, de alguma forma, fazem uso da área de inteligência artificial.

Os serviços podem possuir duas classificações, desse modo, um serviço como o *chat*, por exemplo, pode ser classificado como um serviço de comunicação e interno.

Figura 12 - O Conceito “Serviços” - parte 1

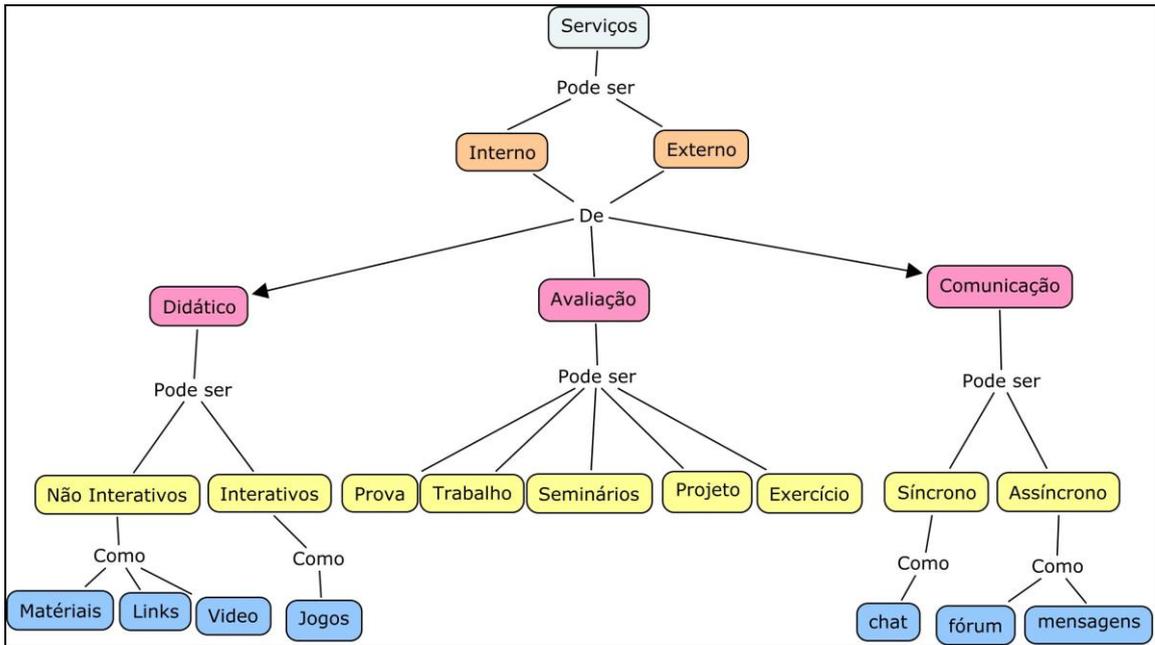
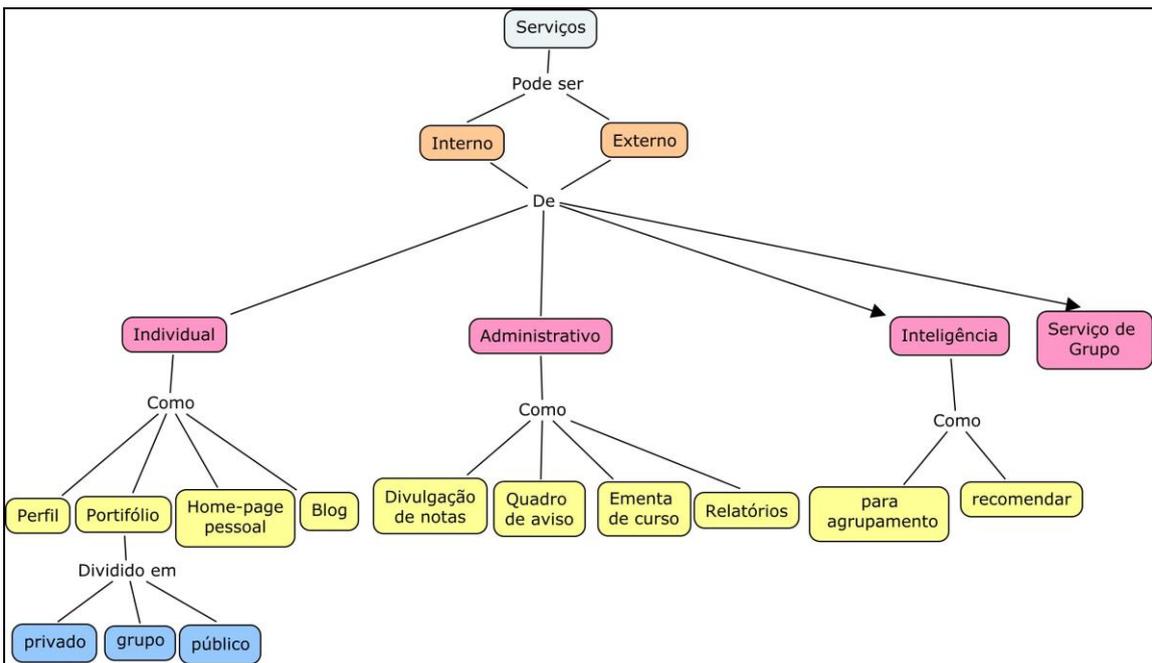


Figura 13 - O Conceito “Serviços” - parte 2



4.2.9. Conceito “grupos”

Este conceito, representado pela Figura 14, trata da possibilidade de formação e organização de grupos, dentro dos cursos, entre os usuários do ambiente.

Segundo Almeida (2001, citado por Almeida, 2003) as interações por meio dos recursos disponíveis no ambiente propiciam as trocas individuais e a constituição de grupos

colaborativos que interagem, discutem problemáticas e temas de interesses comuns, pesquisam e criam produtos ao mesmo tempo em que se desenvolvem.

Neste conceito são considerados quatro aspectos:

i) **Definição do Grupo:** criação do grupo que pode ser classificado em grupo:

- **Simple:** grupos que não possuem subgrupos;
  - **Moderado:** grupos criados pelo professor;
  - **Espontâneo:** grupos que são criados por vontade dos participantes e monitorados pelo professor. Exemplo: alunos que queiram montar um grupo de estudo;
- **Hierárquico:** grupos que possuem subgrupos;
  - **Moderado:** grupos criados pelo professor;
  - **Espontâneo:** grupos que são criados por vontade dos participantes e monitorados pelo professor. Exemplo: alunos que queiram montar um grupo de estudo;
  - **Grupo Geral:** Seria o maior grupo dentro do ambiente, englobando todos os outros grupos.
  - **Grupos isolados:** Seriam grupos que possuem um espaço reservado dentro do ambiente e só podem ser vistos por os outros se for permitido.
  - **Conjunto de grupos:** Seria a junção de grupos isolados na formação de outro grupo.

ii) **Controle de Visibilidade:** definição de visibilidade entre os membros do grupo, por exemplo, definir que apenas os integrantes do grupo A podem ter acesso aos materiais disponíveis no grupo. Esse controle pode ocorrer de duas formas:

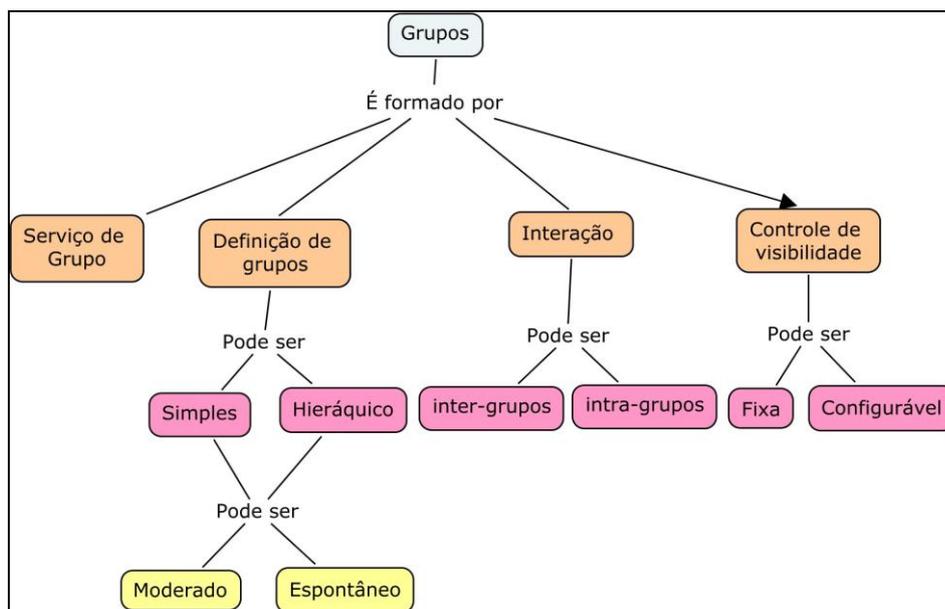
- **Fixa:** a visibilidade é definida na criação do grupo, não podendo ser alterada;
- **Configurável:** apesar de a visibilidade ser definida no momento da criação do grupo, ela pode ser alterada posteriormente.

iii) **Interação:** possibilidade de interação, cooperação e colaboração entre grupos. Pode ocorrer de forma:

- **Inter-grupos:** interação com grupos e subgrupos.
- **Intra-grupos:** interação entre subgrupos do mesmo grupo.

iv) **Grupos Externos:** possibilidade de integração com grupos da web externos ao ambiente.

Figura 14 - O Conceito Grupos



### 4.3. O Mapa Conceitual para o Ator Professor

Este mapa é derivado do mapa geral, mas que está relacionado com a visão do ator “professor” no AVA. Assim, integram o mapa apenas os conceitos, que podem ser disponibilizados pelo ambiente para que os professores possam utilizá-los.

A concepção do mapa conceitual para o ator professor ocorreu a partir de uma pesquisa realizada em duas etapas. A primeira delas foi uma análise exploratória nos ambientes Moodle, Amadeus, Eureka, Claroline, Sakai e Tidia-ae. Nesta primeira etapa:

- i) Foi criado um *login* para o usuário professor em cada um dos ambientes;
- ii) Foram testados e utilizados cada serviço e recurso disponibilizado no ambiente para ator professor;
- iii) Por fim, foram analisadas as funções específicas de cada serviço ou recurso para o professor, por exemplo, um serviço “tarefa” tem uma função para o ator professor, que poder ser avaliar o aluno, e outra função para o ator aluno, que é ser avaliado.

Assim, com base no mapa conceitual geral, foi gerada a primeira versão do mapa conceitual para o professor.

Na segunda etapa foram realizadas algumas entrevistas com professores das áreas ciência da computação e educação dos cursos a distancia. Nesta entrevista, foi questionada ao professor a importância de cada conceito do mapa para o ator professor, para que assim pudesse ser obtida a versão do mapa conceitual para o professor. Através da aplicação de

questionário e com a apresentação, da primeira versão do mapa, a cada entrevistado obteve-se a versão final mapa apresentado nas Figuras 15 e 16.

Figura 15 - O Mapa Conceitual para o professor - parte 1

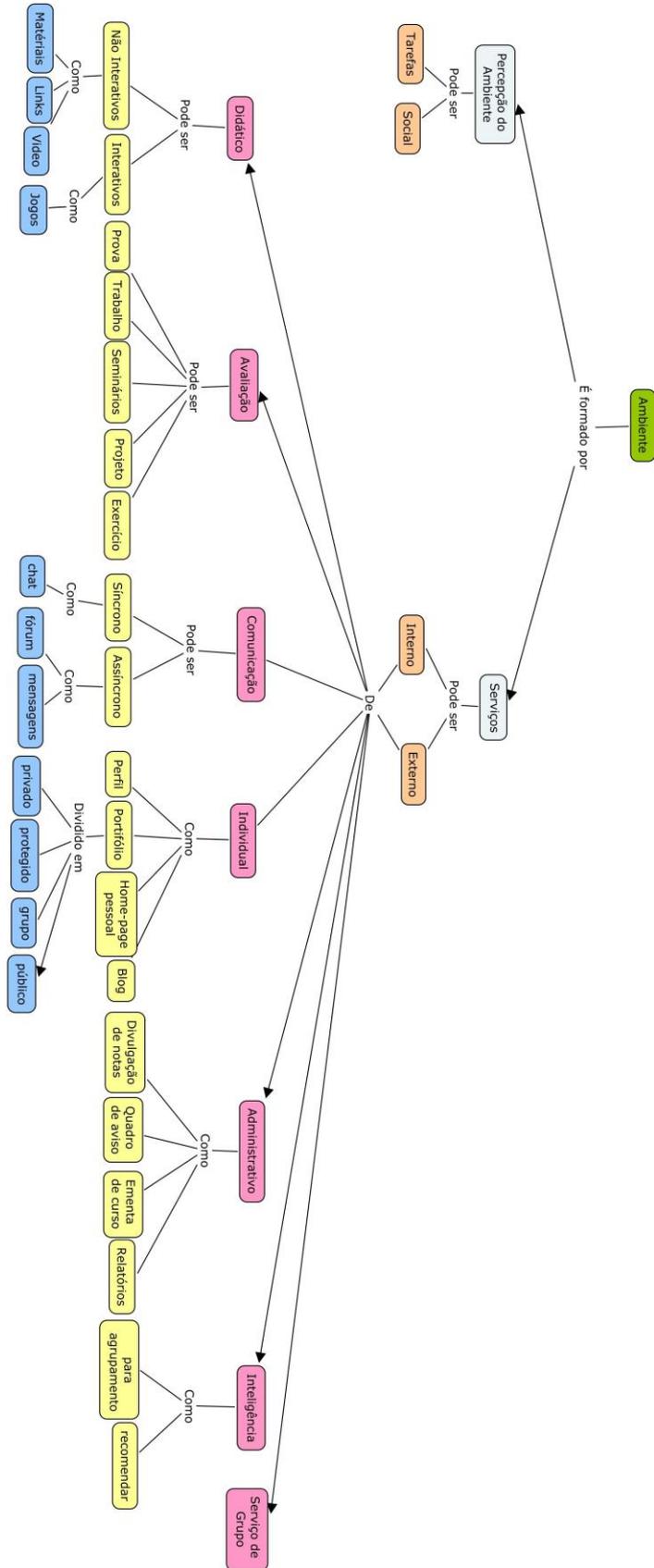
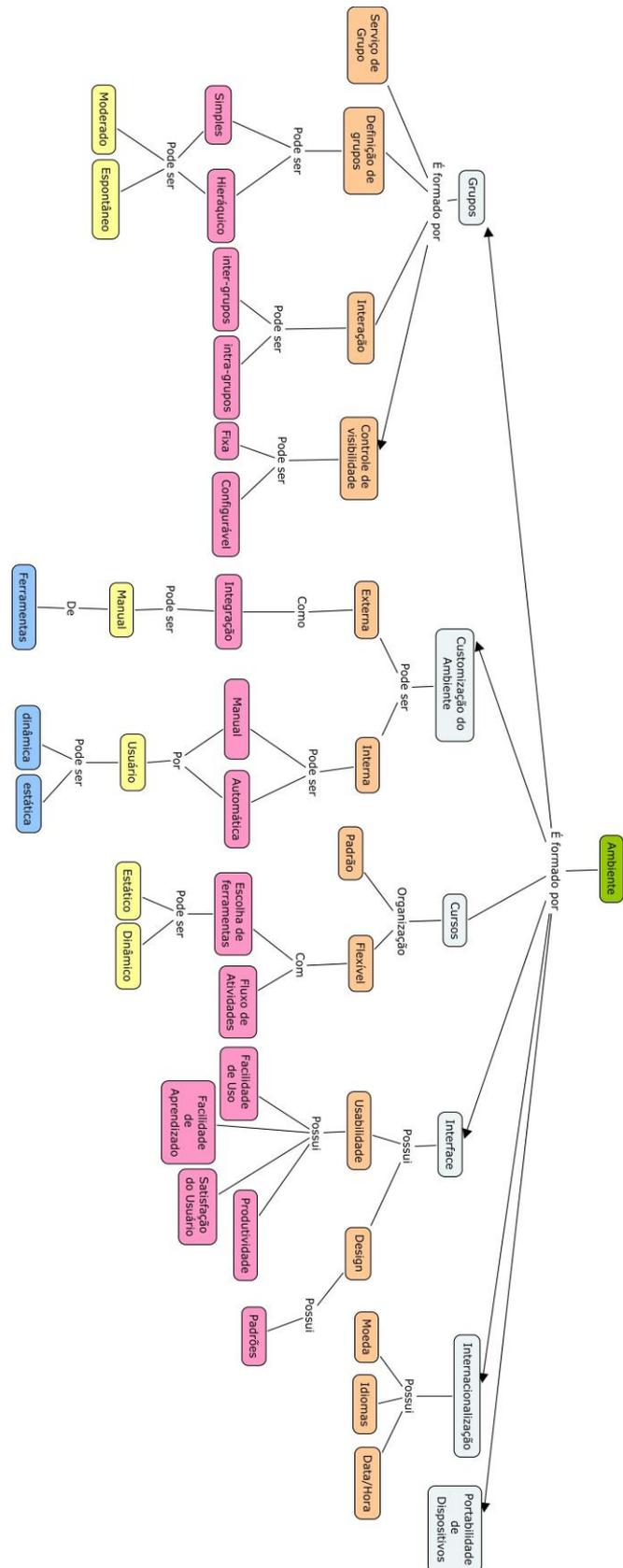


Figura 16 - O Mapa Conceitual para o professor - parte 2



Para o ator professor um dos conceitos mais importantes é o conceito “cursos”. Assim, para o professor, o ambiente pode favorecer a organização e estruturação dos cursos, podendo ocorrer de forma padrão ou flexível. Se houver flexibilidade, o professor poder escolher os serviços utilizados no curso seja de forma estática ou dinâmica. Além de poder determinar um fluxo para suas atividades, pois muitas vezes uma atividade é pré-requisito para outra e isso pode até mesmo ser feito estabelecendo-se uma data de início e término das atividades. Esse fluxo também pode ser flexível, ou seja, o professor pode alterar a ordem, por exemplo, colocando uma atividade que, inicialmente, seria realizada no início do curso, para ser realizada no final.

O conceito “percepção do ambiente” para o professor, pode ser dividido em percepção social e percepção de tarefas. O subconceito percepção social busca averiguar se o ambiente expõe para o usuário aluno informações dos demais usuários, como, por exemplo, indicar se existe alguém online ou se os alunos que estão online no momento. Assim, busca-se analisar se no ambiente são respondidas perguntas como: i) Quem está online? ii) Que está fazendo? iii) O tutor está online?

O subconceito percepção de tarefas busca mostrar ao usuário professor as atividades e tarefas que o mesmo tem pendente, ou seja, que ainda não foram corrigidas ou postadas. Essas tarefas podem ser classificadas como: individuais, aquelas atividades e tarefas que os alunos devem realizar individualmente, ou em grupo, aquelas atividades e tarefas que devem os alunos devem realizar pelos grupos de alunos. Assim, busca-se analisar se no ambiente são respondidas perguntas como: i) Existem atividades pendentes? ii) Quanto falta para concluir determinada atividade?

No conceito customização do ambiente, são abordadas todas as mudanças que podem ocorrer na organização, conteúdo e interface do ambiente para o professor. Essa customização pode ser especificada como interna ou externa. A customização externa refere-se à possibilidade do professor poder integrar alguma ferramenta ao ambiente. A interna é a possibilidade do professor manualmente customizar o ambiente de acordo com suas preferências, podendo ser uma customização estática ou dinâmica. Há também a customização interna automática, que ocorre quando o ambiente percebe o professor e adapta-se às preferências do mesmo e pode ser dinâmica ou estática.

O conceito “portabilidade de dispositivos”, busca identificar se o ambiente permite que o professor acesse seus cursos através de diferentes dispositivos, como a TV Digital, computador, celular e etc. Já a internacionalização é a capacidade do ambiente oferecer suporte para que o professor possa acessar seus cursos em diferentes idiomas, diferentes moedas e data/hora.

Dois conceitos merecem um destaque maior, quando analisados para o professor. São eles:

### 1. Serviços

- **Interno:** Serviço interno pode ser qualquer serviço que o professor acesse no AVA, desde que ele seja nativo do ambiente, ou seja, pode ser serviço didático, administrativo, de grupo e etc., no entanto precisa ter sido desenvolvido junto ao ambiente.
- **Externo:** Serviço externo pode ser qualquer serviço do AVA, desde que ele seja acoplado ao ambiente pelo professor, ou seja, pode ser serviço didático, administrativo, de grupo e etc., no entanto precisa ter sido desenvolvido fora do ambiente.

1. **Administrativo:** permitem que professor, possa gerenciar o curso;

Exemplos:

- Agenda do curso;
- Quadro de avisos;
- Ementa de curso;
- Divulgação de notas;
- Controle do progresso do aluno;

2. **Didático:** são disponibilizados pelo professor com a finalidade de viabilizar o aprendizado e podem ser classificados como:

1. **Interativo:** São serviços dinâmicos;
2. **Não Interativo:** São serviços estáticos;

#### 3. Tipos:

- Transparências;
- Referências na Web;
- Glossário indexado;
- Facilidade para anotações de aluno;
- Material de referência do curso;

- Quadro branco compartilhado e interativo;
  - Documentos multimídia;
  - Imagens;
  - Som;
  - Vídeo;
  - Jogos;
3. **Avaliação:** são serviços disponibilizados pelo professor que possuem como finalidade avaliar o desempenho do aluno;
- Provas;
  - Auto-avaliação;
  - Exercícios;
  - Tarefas;
4. **Comunicação:** são todos serviços que, de alguma forma, o professor pode utilizar para estabelecer uma comunicação com os alunos. Esta comunicação poderá ocorrer de forma:
1. **Síncrona:** quando é necessário que tanto o usuário emissor quanto o usuário receptor estejam conectados no AVA ao mesmo tempo;
  2. **Assíncrona:** quando não é necessário que o usuário emissor e o usuário receptor estejam conectados no AVA ao mesmo tempo;
  3. **Tipos:**
    - Comunicação individual;
    - Notificações;
    - Comunicação grupal;
5. **Grupo:** oferecem suporte a formação e organização de grupos.
- Discussão;
  - Compartilhamento de materiais;
6. **Individuais:** são os serviços que o professor pode utilizar, de modo pessoal, para armazenar suas informações e materiais.
- Home-pages pessoais;
  - Arquivos de imagem indexados;
  - Indexação e busca automática;
  - Perfis dos professores;
  - **Portfólio:** para existir um controle de visibilidade e um espaço reservado dentro do ambiente, o portfólio deveria ser dividido em:

- **Privado:** materiais que somente o usuário tem acesso. No caso do aluno, materiais inacabados aos quais ele não deseja que o professor tenha acesso. No caso do professor, materiais que ele ainda vai passar para os alunos.
  - **Protegido:** materiais que só o professor e o aluno têm acesso;
  - **Grupo:** espaço reservado ao compartilhamento de materiais pelo grupo;
  - **Público:** materiais que todos dentro do ambiente podem visualizar.
7. **Inteligência:** serviços utilizados pelo professor que, de alguma forma, fazem uso da área de inteligência artificial.

Os serviços podem possuir duas classificações, desse modo, um serviço como o *chat*, por exemplo, pode ser classificado como um serviço de comunicação e interno.

## 2. Grupos

Este conceito trata da possibilidade de formação e organização de grupos entre os usuários do ambiente. Neste conceito são considerados quatro aspectos:

1. **Grupos externos:** integração de grupos que estão fora do ambiente;
  2. **Definição de grupos:** criação do grupo que pode ser classificado em:
    1. **Simple:** grupos que não possuem subgrupos;
      1. **Moderado:** grupos criados pelo professor;
      2. **Espontâneo:** grupos que são criados por vontade dos participantes e monitorados pelo professor. Exemplo: alunos que queiram montar um grupo de estudo;
    2. **Hierárquico:** grupos que possuem subgrupos;
      1. **Moderado:** grupos criados pelo professor;
      2. **Espontâneo:** grupos que são criados por vontade dos participantes. Exemplo: alunos que queiram montar um grupo de estudo;
- Obedecendo a uma hierarquia os grupos podem ser formados dentro do ambiente com as seguintes características:
- **Grupo Geral:** Seria o maior grupo dentro do ambiente, englobando todos os outros grupos.
  - **Grupos isolados:** Seriam grupos que possuem um espaço reservado dentro do ambiente e só podem ser vistos por os outros se for permitido.
  - **Conjunto de grupos:** Seria a junção de grupos isolados na formação de

outro grupo.

3. **Controle de visibilidade:** definição de visibilidade entre os membros do grupo, por exemplo, definir que apenas os integrantes do grupo A podem ter acesso aos materiais disponíveis no grupo. Esse controle pode ocorrer de duas formas:
  1. **Fixo:** a visibilidade é definida na criação do grupo, não podendo ser alterada;
  2. **Configurável:** apesar da definição da visibilidade ser no momento da criação do grupo, e ela pode ser alterada posteriormente;
4. **Interação:** possibilidade de interação, cooperação e colaboração entre grupos;
  1. **Inter-grupos:** interação com grupos e subgrupos;
  2. **Intra-grupos:** interação entre subgrupos;

#### 4.4. O Mapa Conceitual para o Ator Aluno

Este mapa também é derivado do mapa geral, mas agora está relacionado com a visão do ator “aluno”. Assim como no mapa para o professor, este é formado apenas pelos conceitos utilizados pelo aluno sejam para responder uma atividade do curso ou até mesmo para configurar seu perfil.

A concepção do mapa conceitual para o ator aluno ocorreu de forma semelhante a do mapa do professor, sendo que este foi feito a partir de uma pesquisa realizada em três etapas. A primeira delas foi uma análise exploratória nos ambientes Moodle, Amadeus, Eureka, Claroline, *Sakai* e Tidia-ae. Nesta primeira etapa:

- i) Foi criado um *login* para o usuário aluno em cada um dos ambientes;
- ii) Foram testados e utilizados cada serviço e recurso disponibilizado no ambiente para ator aluno;
- iii) Por fim, foram analisadas as funções específicas de cada serviço ou recurso para o aluno, por exemplo, um serviço “fórum” tem uma função para o ator professor, que poder ser avaliar o aprendizado do aluno, e outra função para o ator aluno, que é aprender através da interação com o mesmo.

Assim, com base no mapa conceitual geral e mapa conceitual para o ator professor, foi gerada a primeira versão do mapa conceitual para o aluno.

Na segunda etapa foram realizadas algumas entrevistas com professores das áreas ciência da computação e educação dos cursos a distancia. Nesta entrevista, foi questionada ao

professor a importância de cada conceito do mapa para o ator aluno, assim obteve-se a segunda versão do mapa conceitual para o aluno.

Para a terceira etapa foram feitas entrevistas com alguns alunos dos cursos a distância. Esta entrevista procedeu-se da seguinte forma, foi apresentado a cada aluno a segunda versão do mapa, para que os mesmos pudessem identificar a real utilidade de cada conceito e subconceitos. Como isso, Através da aplicação de questionário e com a apresentação da primeira versão do mapa, a cada entrevistado, alcançou-se a versão final mapa apresentado nas Figuras 17 e 18.

Figura 17 - O Mapa Conceitual para o aluno - parte 1

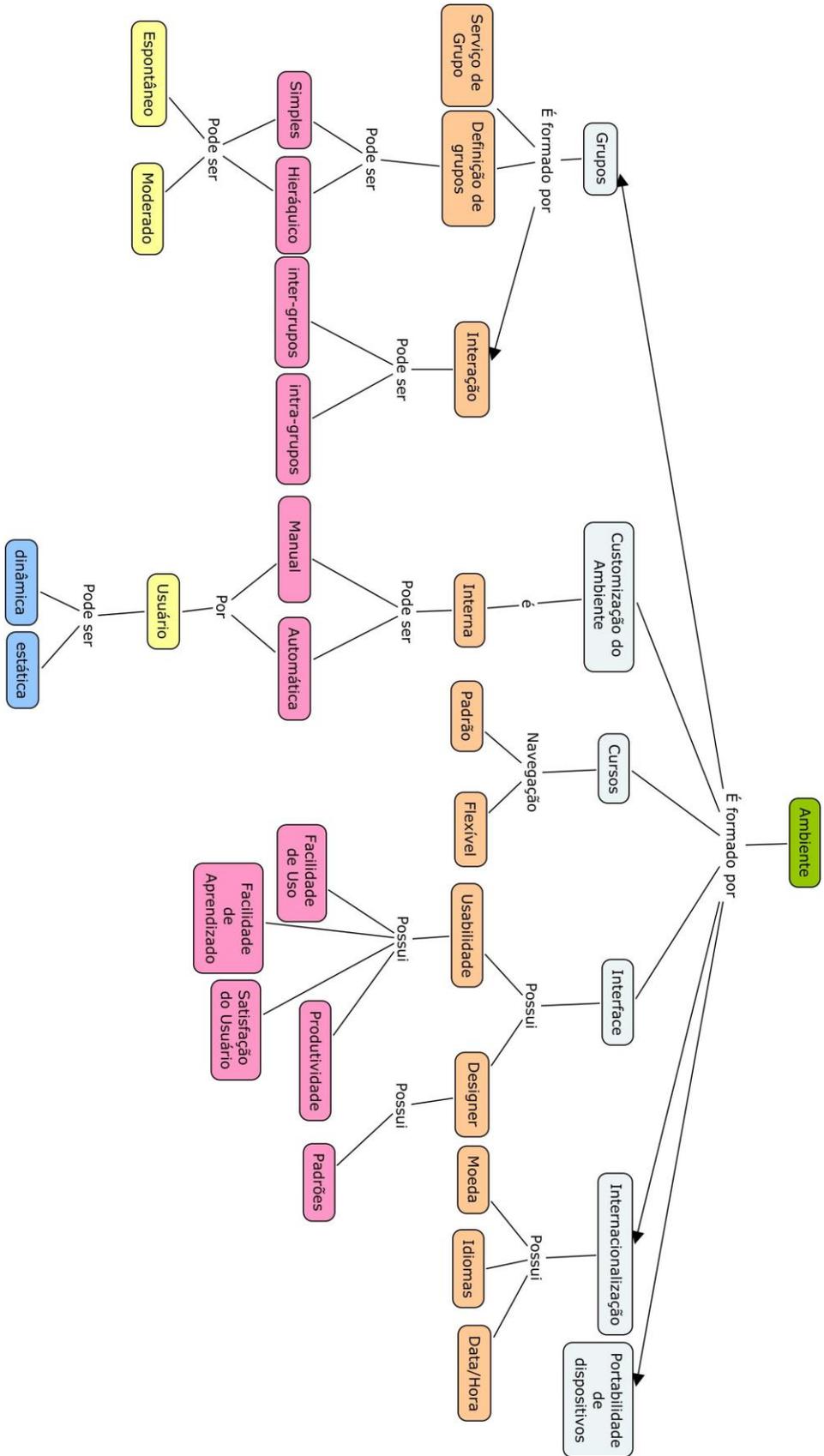
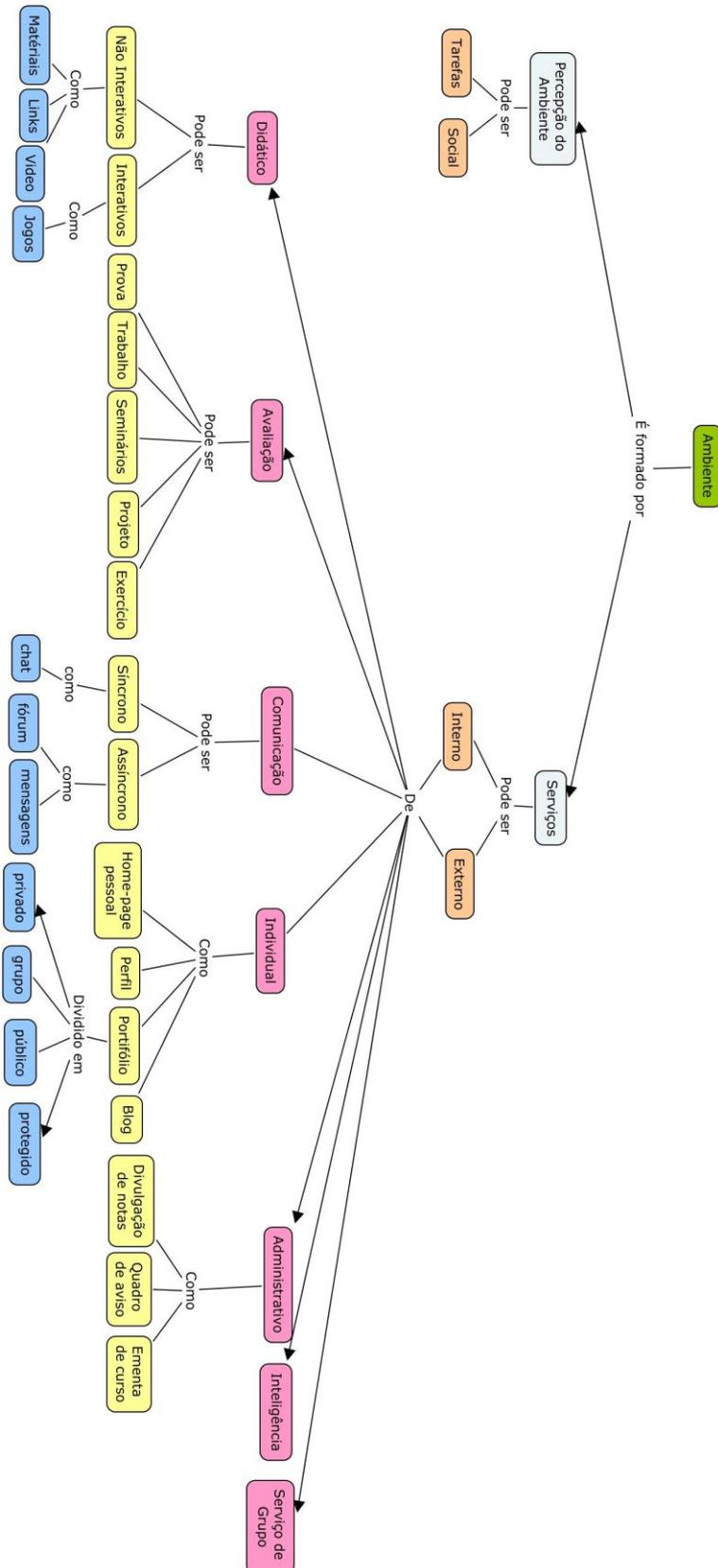


Figura 18 - O Mapa Conceitual para o aluno - parte 2



Assim como para o ator professor, para o aluno, o conceito mais importante é o conceito “cursos”. Com isso, para o aluno, o ambiente pode favorecer a organização e

estruturação dos cursos, podendo ocorrer de forma padrão ou flexível. A forma padrão é uma estrutura que oferece aos alunos cursos apresentados de forma padronizada, com os mesmos tipos de atividades. Por exemplo, determinar que todos os cursos devessem conter os serviços fórum, chat e tarefas. Assim, o usuário aluno não terá cursos com estruturas distintas, pois o ambiente não permite.

A forma flexível é uma estrutura onde os cursos são flexíveis para os alunos, no que se refere à possibilidade dos alunos estudarem em cursos com tipos de atividades diferentes. Por exemplo, um curso pode conter os serviços fórum, chat e tarefas e outro pode não conter o fórum.

O conceito “percepção do ambiente” para o aluno, pode ser dividido em percepção social e percepção de tarefas. O subconceito percepção social busca averiguar se o ambiente expõe para o usuário aluno informações dos demais usuários, como, por exemplo, indicar se o professor ou o tutor está online ou os alunos que estão online no momento. Assim, busca-se analisar se no ambiente são respondidas perguntas como: i) Quem está online? ii) Que está fazendo? iii) O tutor está online?

O subconceito percepção de tarefas busca mostrar ao usuário aluno as atividades e tarefas que o mesmo tem pendente, ou seja, que ainda não foram respondidas ou enviadas. Essas tarefas podem ser classificadas como: individuais, aquelas atividades e tarefas que o aluno fez de forma individual, ou em grupo, aquelas atividades e tarefas que o aluno fez em grupo. Assim, busca-se analisar se no ambiente são respondidas perguntas como: i) Existem atividades pendentes? ii) Quanto falta para concluir determinada atividade?

No conceito customização do ambiente, são abordadas todas as mudanças que podem ocorrer na organização, conteúdo e interface do ambiente para o aluno. Essa customização pode ser especificada como interna que é a possibilidade do aluno manualmente customizar o ambiente de acordo com suas preferências, podendo ser uma customização estática ou dinâmica. Há também a customização interna automática, que ocorre quando o ambiente percebe o aluno e adapta-se às preferências do mesmo e pode ser dinâmica ou estática.

O conceito “portabilidade de dispositivos”, busca identificar se o ambiente permite que o aluno acesse seus cursos através de diferentes dispositivos, como a TV Digital, computador, celular e etc. Já o conceito “internacionalização” é a capacidade de o ambiente oferecer suporte para que o professor possa acessar seus cursos em diferentes idiomas, diferentes moedas e data/hora.

Os dois componentes a seguir merecem um detalhamento maior, também, quando analisados para o professor. São eles:

### 1. Serviços

- **Interno:** Serviço interno pode ser qualquer serviço que o aluno acesse no AVA, desde que ele seja nativo do ambiente, ou seja, pode ser serviço didático, administrativo, de grupo e etc., no entanto precisa ter sido desenvolvido junto ao ambiente.
- **Externo:** Serviço externo pode ser qualquer serviço do AVA, desde que ele seja acoplado ao ambiente pelo professor, ou seja, pode ser serviço didático, administrativo, de grupo e etc., no entanto precisa ter sido desenvolvido fora do ambiente e os alunos utilizarem.
  1. **Administrativo:** permitem que o aluno possa acompanhar o desenvolvimento do curso. Exemplos: agenda do curso, quadro de avisos, ementa de curso e divulgação de notas;
  2. **Didático:** são utilizados pelos alunos para viabilizar o seu aprendizado e podem ser classificados como:
    1. **Interativo:** São serviços dinâmicos;
    2. **Não Interativo:** São serviços estáticos;
    3. **Tipos:**
      - Transparências;
      - Referências na Web;
      - Glossário indexado;
      - Facilidade para anotações de aluno;
      - Material de referência do curso;
      - Quadro branco compartilhado e interativo;
      - Documentos multimídia;
      - Imagens;
      - Som;
      - Vídeo;
      - Jogos;
  3. **Avaliação:** são serviços disponibilizados pelo professor e utilizados pelos alunos avaliar o seu desempenho no curso; Exemplos: provas, auto-avaliação, exercícios e tarefas.
  4. **Comunicação:** são todos serviços que, de alguma forma, o aluno pode utilizar

para estabelecer uma comunicação tanto com os demais alunos quanto com os professores e tutores. Esta comunicação poderá ocorrer de forma síncrona ou assíncrona;

5. **Grupo:** oferecem suporte a formação e organização de grupos. Podem ser grupos de discussão, compartilhamento de materiais e etc;
6. **Individuais:** são os serviços que o aluno pode utilizar, de modo pessoal, para armazenar suas informações e materiais.
  - Home-page pessoais;
  - Arquivos de imagem indexados;
  - Indexação e busca automática;
  - Perfil;
  - **Portfólio:** para existir um controle de visibilidade e um espaço reservado dentro do ambiente, o portfólio deveria ser dividido em:
    - **Privado:** materiais que somente o usuário tem acesso. No caso do aluno, materiais inacabados aos quais ele não deseja que o professor tenha acesso. No caso do professor, materiais que ele ainda vai passar para os alunos.
    - **Protegido:** materiais que só o professor e o aluno têm acesso;
    - **Grupo:** espaço reservado ao compartilhamento de materiais pelo grupo;
    - **Público:** materiais que todos dentro do ambiente podem visualizar.
7. **Inteligência:** serviços utilizados pelo aluno que, de alguma forma, fazem uso da área de inteligência artificial.

Os serviços podem possuir duas classificações, desse modo, um serviço como o *chat*, por exemplo, pode ser classificado como um serviço de comunicação e interno.

## 2. Grupos

Este conceito trata da possibilidade de formação e organização de grupos entre os usuários do ambiente. Neste conceito são considerados quatro aspectos:

1. **Grupos externos:** integração de grupos que estão fora do ambiente;
2. **Definição de grupos:** criação do grupo que pode ser classificado em:
  1. **Simple:** grupos que não possuem subgrupos;
  1. **Moderado:** grupos criados pelo professor, mas que os alunos participam;

2. **Espontâneo:** grupos que são criados por vontade dos participantes e monitorados pelo professor. Exemplo: alunos que queiram montar um grupo de estudo;
2. **Hierárquico:** grupos que possuem subgrupos;
  1. **Moderado:** grupos criados pelo professor, mas que os alunos participam;
  2. **Espontâneo:** grupos que são criados por vontade dos participantes. Exemplo: alunos que queiram montar um grupo de estudo;
    - **Grupo Geral:** Seria o maior grupo dentro do ambiente, englobando todos os outros grupos.
    - **Grupos isolados:** Seriam grupos que possuem um espaço reservado dentro do ambiente e só podem ser vistos por os outros se for permitido.
    - **Conjunto de grupos:** Seria a junção de grupos isolados na formação de outro grupo.
3. **Interação:** possibilidade de interação, cooperação e colaboração entre grupos;
  1. **Inter-grupos:** interação com grupos e subgrupos;
  2. **Intra-grupos:** interação entre subgrupos;

## 5. AVALIAÇÃO DE AMBIENTES VIRTUAIS DE APRENDIZAGEM

Neste capítulo serão discutidas as análises comparativas realizadas em seis ambientes. Assim, será apresentada uma versão do mapa conceitual, para cada AVA. Ao final será apresentada uma tabela comparando os ambientes entre si, com relação a algumas principais características extraídas do mapa conceitual.

### 5.1. Amadeus

“O ambiente virtual de aprendizagem Amadeus tem origem em um conjunto de pesquisas acadêmicas na área de Interação Humano Computador e Tecnologia Educacional”. (MELO et al, 2011, p.30)

O Amadeus (2011) é um AVA desenvolvido por pesquisadores da UFPE – Universidade Federal de Pernambuco e “visa o desenvolvimento de um sistema de gestão da aprendizagem de segunda geração, baseado no conceito de *blended learning*” (GABARDO; QUEVEDO e ULBRICHT, 2010, 70). A marca registra deste ambiente é a portabilidade, já que ele permite criar situações de aprendizagem em diversos contextos utilizando uma variedade de mídias. “O projeto permite estender as experiências adquiridas de usuários de educação à distância para diversas plataformas (Internet, *desktop*, celulares, *PDA*s, e futuramente TV Digital) de forma integrada e consistente” (GABARDO; QUEVEDO e ULBRICHT, 2010, 70). As Figuras 19 e 20 representam o mapa conceitual para este ambiente.

Atualmente o LMS Amadeus é distribuído sob uma licença de Software Livre e desde março de 2009, passou a integrar o PSPB2 – Portal do Software Público Brasileiro. A comunidade do Projeto Amadeus no PSPB conta com mais de 5000 membros e com colaborações internacionais de países como: Irlanda, Chile, Alemanha e França. (MELO ET al, 2011, p.30)



- Professor: responsável pela organização e gerenciamento da plataforma nas diversas disciplinas.
- Estudante: estudantes/alunos.

### 5.1.2. Conceito “interface” no Amadeus

Para o conceito “interface”, o Amadeus mostrou-se bem em vários subconceitos. No conceito usabilidade, que considera a qualidade de interação com os usuários, ele se mostrou fácil de aprender, pois os caminhos são curtos, os ícones são sugestivos e a interface é leve, navegando-se uma ou duas vezes no ambiente já é possível utilizá-lo sem auxílio de manual.

O ambiente também é fácil usar, pois não requer um grande esforço, sendo bem intuitivo. É um ambiente agradável de trabalhar. No entanto não é produtivo, pois para realizar algumas atividades o aluno precisa utilizar ferramenta que não são oferecidas pelo ambiente e depois postá-la no mesmo. No design, ele possui harmonia entre cores utilizadas, tipos e formatos de letras e caixas de diálogos que se mantêm ao longo de todo o ambiente.

### 5.1.3. Conceito “cursos” no Amadeus

No mapa geral, este conceito busca analisar o suporte que o ambiente oferece para a estrutura e organização de cursos. No Amadeus o professor deve estrutura o curso em módulos onde são inseridos os serviços, por exemplo, exercícios e fóruns, sendo assim, ele tem a liberdade de acrescentar os recursos e serviços, que desejar. A organização e estruturação dos cursos, no mapa geral, podem ocorrer das seguintes formas:

- a) Padrão, quando ela é fixa para todo e qualquer curso, mas o Amadeus não implementa esse tipo de estrutura e organização.
- b) Flexível, quando na estrutura do curso existe a flexibilidade tanto de escolher os serviços que serão utilizados no curso, como também o fluxo das atividades. O Amadeus permite que os cursos desenvolvidos nele sejam flexíveis.

No mapa geral, a flexibilidade de cursos é composta pelos conceitos:

- i) Escolha de serviços: o professor tem autonomia para determinar, por exemplo, se vai utilizar um fórum ou chat no curso, essa escolha pode ser feita de duas formas: estática ou dinâmica. No Amadeus, a escolha de serviços é dinâmica, pois nele o professor pode acrescentar durante o curso diversos recursos, como por exemplo, fórum.
- ii) Fluxo de atividades: Cada professor pode escolher a sequência das ferramentas que serão trabalhadas. No Amadeus, o professor pode determinar esse fluxo de atividades:

Cada usuário professor pode escolher a sequência das atividades que serão trabalhadas. Por exemplo:

- A divisão do curso em módulos, já é considerada um fluxo para as atividades;
- Escolher a data de início e término das atividades;

#### 5.1.4. Conceito “percepção do ambiente” no Amadeus

No mapa geral, este conceito, são tratados aspectos de percepção de tarefas e atividades, individuais e do grupo, como também a percepção social dentro do ambiente. Sendo subdividido em percepção social e percepção de tarefas.

O Amadeus, para a percepção social, oferece um recurso, o item Usuários Online, que lista os usuários que estão naquele momento no ambiente. Oferece outro recurso também, no item colegas de sala, onde o professor pode visualizar todos os seus cursos com os seus respectivos alunos. Dentro de cada curso existe o item “participantes”, onde o professor pode visualizar os alunos daquele curso;

Para percepção de tarefas, no Amadeus foi encontrado um recurso chamado Tarefas Pendentes, que indica as atividades pendentes para um usuário professor, por exemplo, alguma atividade que ele deve corrigir.

#### 5.1.5. Conceito “customização do ambiente” no Amadeus

Este conceito, no mapa geral, abrange todas as mudanças que podem ocorrer na organização, conteúdo e interface do ambiente. Essa customização pode ser especificada como, interna com adaptação manual ou automática para usuário, de forma estática ou dinâmica e adaptação manual ou automática para instituições e departamentos; ou externa, com a integração manual ou automática de ferramentas e AVA. O Amadeus, por ser de código aberto, permite a integração com outras ferramentas de forma manual. Além de permitir uma customização interna, automática e dinâmica por usuário, pois o Amadeus possui ambientes diferentes, para cada usuário, que pode sofrer alteração no decorrer do curso.

#### 5.1.6. Conceito “portabilidade para dispositivos” no Amadeus

No mapa geral, a portabilidade para dispositivos indica se o ambiente oferece suporte para ser acessado a partir de diferentes dispositivos. O Amadeus permite acesso através de múltiplos dispositivos, TV Digital e mídias para celulares. Possui um módulo de integração com a TV Digital.

### 5.1.7. Conceito “internacionalização” no Amadeus

Este conceito, no mapa geral, refere-se à capacidade do ambiente de oferecer suporte para que seus cursos possam ser acessados em diferentes idiomas e possua formato para diferentes moedas e data/hora. O Amadeus não informa a implementação deste conceito.

### 5.1.8. Conceito “serviços” no Amadeus

No mapa geral, o conceito serviços, trata dos recursos utilizados no desenvolvimento do curso, podendo ser classificados como, serviços internos ou externos. E esses ainda podem ser serviços administrativos, didáticos, de avaliação, de comunicação, individuais, de grupo e de inteligência.

Sendo assim, o Amadeus possui serviços para todas as categorias. No entanto, serviços externos só foram encontrados nas mídias para celular. Para os serviços internos encontramos nos administrativos, os serviços que permitem que professor, possa gerenciar o seu curso, por exemplo, ementa de curso e divulgação de notas. Nos internos didáticos, o ambiente implementa tanto os interativo, como jogos, fórum e enquetes, quanto os não interativo, como entrega de material e referências na web – links, entre outros como transparências, material de referência do curso, documentos multimídia, vídeo e questionários.

Os serviços internos para avaliação são tarefas, fórum, entrega de material. Para os serviços internos de comunicação o Amadeus só implementa serviços assíncronos como, agenda do curso, fóruns e últimas notícias. Como serviço interno de grupo há apenas o vídeo em grupo. Nos serviços individuais há os perfis dos usuários.

### 5.1.9. Conceito “grupos” no Amadeus

Este conceito, no mapa geral, trata da possibilidade de formação e organização de grupos entre os usuários do ambiente dentro do curso. O Amadeus não implementa este conceito.

## 5.2. Moodle

Segundo Ribeiro e Mendonça (2007), o Moodle é uma plataforma Open Source, podendo ser instalado, utilizado, modificado e mesmo distribuído. Tem como objetivo o gerenciamento de aprendizado e de trabalho colaborativo em ambiente virtual, permitindo a

criação e administração de cursos on-line, grupos de trabalho e comunidades de aprendizagem.

Conforme Oliveira, Munhoz e Carneiro (2011), este ambiente vem sendo utilizado em todo o mundo por diversas instituições, possuindo uma grande comunidade cujos membros estão envolvidos em atividades que abrangem desde correções de erros e o desenvolvimento de novas ferramentas à discussão sobre estratégias pedagógicas de utilização do ambiente e suas interfaces. Como qualquer outro Learning Management System - LMS, o MOODLE (2011) dispõe de um conjunto de ferramentas que podem ser selecionadas pelo professor de acordo com seus objetivos. As Figuras 21 e 22 representam o mapa conceitual para este ambiente.

Figura 20 - Mapa Conceitual do Ambiente Moodle – parte 1

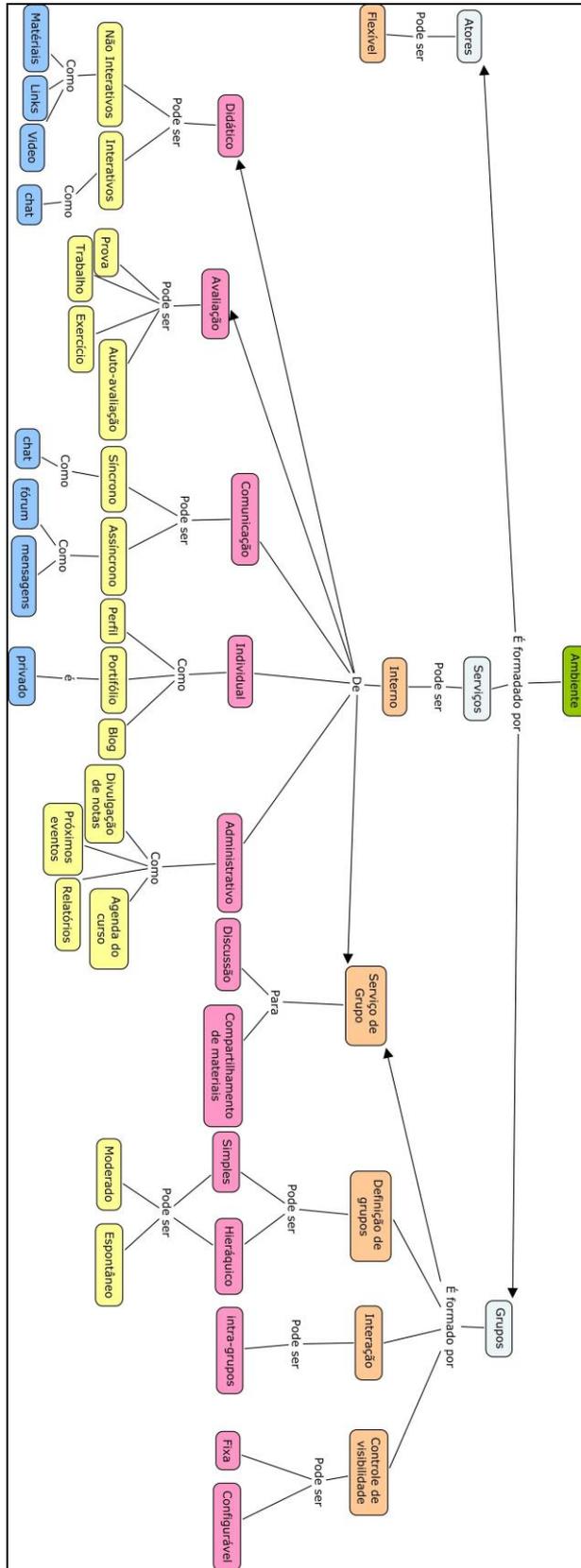
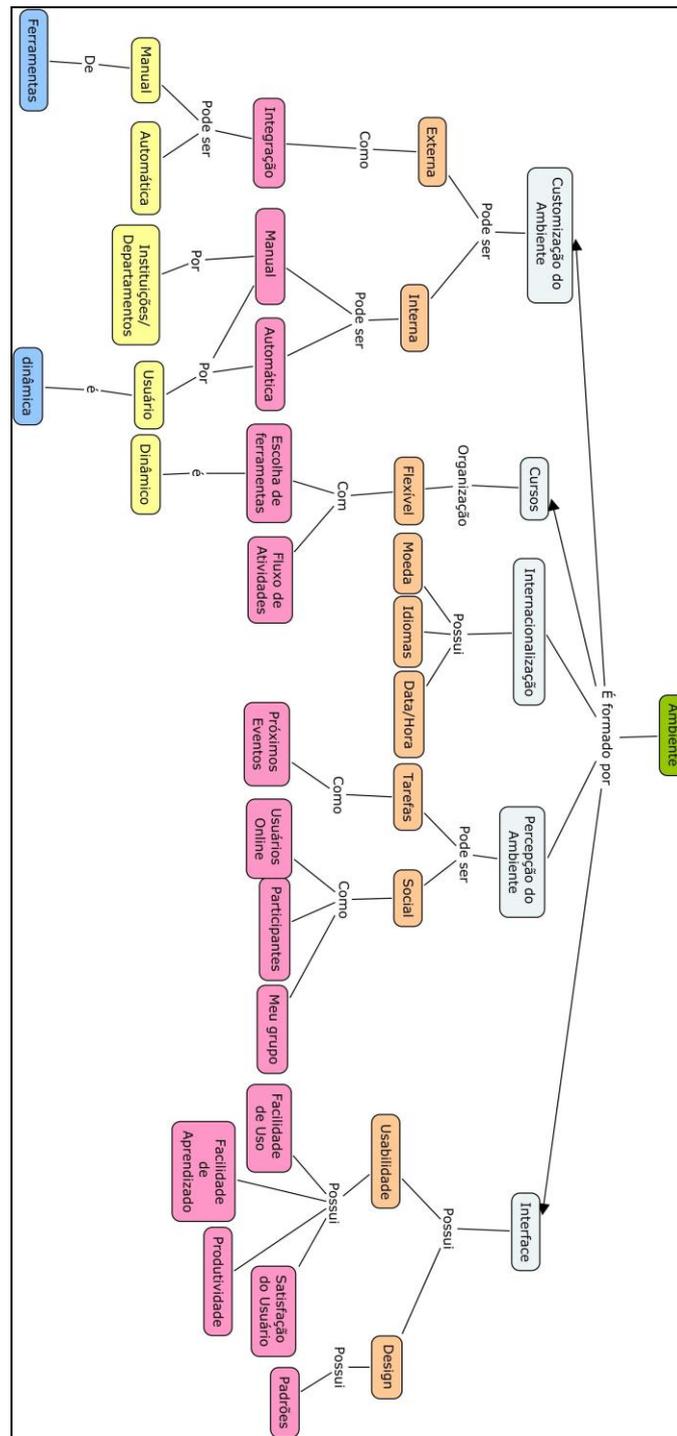


Figura 21 - Mapa Conceitual do Ambiente Moodle – parte 2



### 5.2.1. Conceito “atores” no Moodle

No mapa geral, este conceito considera a existência de atores que interagem com o ambiente. Esses atores podem existir de duas formas: fixa ou flexível. Sendo que no Moodle os atores são flexíveis, ou seja, podem ser acrescentados e removidos pelo usuário final. No entanto temos alguns básicos como:

- Administrador: responsável pela administração, configuração, etc.
- Professor: responsável pela organização e gerenciamento da plataforma nas diversas disciplinas.
- Estudante: estudantes/alunos.

### 5.2.2. Conceito “interface” no Moodle

A interface principal do Moodle é voltada para o favorecimento do enfoque nas atividades em andamento no curso. É composta por caixas localizadas nas laterais da janela.

No conceito usabilidade, que considera a qualidade de interação com os usuários, ele não se mostrou fácil de aprender, pois os caminhos são muito longos, os ícones não são sugestivos e na interface são disponibilizadas muitas informações.

O ambiente também não é fácil usar, pois requer um grande esforço cognitivo. Não é um ambiente agradável de trabalhar e nem é produtivo, pois para realizar algumas atividades o aluno precisa utilizar ferramenta que não são oferecidas pelo ambiente e depois postá-la no mesmo. No design, ele não possui harmonia entre cores utilizadas, mas tipos e formatos de letras e caixas de diálogos se mantêm ao longo de todo o ambiente.

### 5.2.3. Conceito “cursos” no Moodle

No mapa geral, este conceito busca analisar o suporte que o ambiente oferece para a estrutura e organização de cursos. Basicamente, no Moodle, o professor pode estruturar o curso em tópicos, onde cada tópico possui seu conjunto de serviços, além disso, podem-se acrescentar os recursos e serviços, existentes no ambiente, que desejar. A organização e estruturação dos cursos, no mapa geral, podem ocorrer das seguintes formas:

a) Padrão, quando ela é fixa para todo e qualquer curso, no entanto o Moodle não implementa esse tipo de estrutura e organização.

b) Flexível, quando na estrutura do curso existe a flexibilidade tanto de escolher os serviços que serão utilizados no curso, como também o fluxo das atividades. O Moodle permite que os cursos desenvolvidos nele sejam flexíveis. Na criação ou edição do curso, ao lado de cada recurso (atividade, fórum) existem botões como: “ mover”, que altera a ordem de exibição, na vertical, do fórum, “mover para direita”, “mover para esquerda”, “atualizar”, “duplicar”, “excluir”, “ocultar” ou “mostrar”, “Designar funções” e outro para atribuir grupos.

No mapa geral, a flexibilidade de cursos é composta pelos conceitos:

- i. Escolha de serviços: o professor tem autonomia para determinar, por exemplo, se vai utilizar um fórum ou chat no curso, essa escolha pode ser feita de duas formas: estática ou dinâmica. No Moodle de forma dinâmica, o usuário professor pode realizar modificações, ou seja, adicionar e remover serviços, a qualquer momento do curso;
- ii. Fluxo de atividades: Cada professor pode escolher a sequência das ferramentas que serão trabalhadas. No Moodle, o professor pode determinar fluxo de atividades, cada usuário professor pode escolher a sequência das ferramentas que serão trabalhadas. Por exemplo:
  - Escolher a data de início e término das atividades;
  - Deixar visível ou esconder uma atividade;
  - Determina a sequência das atividades e essa sequência deve ser flexível, ou seja, o usuário professor pode alterar a ordem, por exemplo, colocando uma atividade que, inicialmente, seria realizada no início do curso, para ser realizada no final.

#### 5.2.4. Conceito “percepção do ambiente” no Moodle

No mapa geral, este conceito, são tratados aspectos de percepção de tarefas e atividades, individuais e do grupo, como também a percepção social dentro do ambiente. Sendo subdividido em: percepção social e percepção de tarefas.

Neste conceito, o Moodle, para a percepção social, oferece o recurso, usuários online, que lista os usuários que estão (ou estiveram) acessando a sala nos últimos cinco minutos. Oferece, também, o recurso “participantes”, onde o usuário tem uma lista rápida de todos os participantes (tutores e alunos) do curso, sendo possível também consultar os seus perfis.

O recurso meu grupo, permite a consulta rápida de qual grupo o usuário pertence, bem como quais são seus colegas de grupo. Esse grupo pode ser usado, a critério do professor ou tutor, para a realização de atividades, discussões temáticas, construção de wikis e etc.

Na percepção de tarefas, o Moodle oferece um recurso que indica as atividades recentes no ambiente e outro que o professor pode visualizar os alunos que estão enviando as atividades.

O recurso próximos eventos disponibiliza uma relação de eventos em ordem cronológica informando inclusive a hora para seu término. Compreende-se por evento qualquer atividade com prazo definido, como: entrega de trabalho, fechamento de fóruns,

inscrição em disciplinas, etc. Os eventos podem estar organizados de acordo com quatro tipos:

- Eventos globais: mostra eventos de todos os ambientes em que o aluno esteja inscrito.
- Eventos de curso: calendário de atividades do curso (disciplinas).
- Eventos do grupo: editados e disponibilizados pelos membros do grupo.
- Eventos do usuário: permite ao usuário inserir um evento na agenda que será visível somente a ele.

#### 5.2.5. Conceito “customização do ambiente” no Moodle

No mapa geral, este conceito abrange todas as mudanças que podem ocorrer na organização, conteúdo e interface do ambiente. Essa customização pode ser especificada como, interna com adaptação manual ou automática para usuário, de forma estática ou dinâmica e adaptação manual ou automática para instituições e departamentos ou externa, com a integração manual ou automática de ferramentas e AVA.

Por ser de código aberto, o Moodle permite a integração com outras ferramentas de forma manual proporciona a integração com “plugins” que podem ser desenvolvidos como recursos ou atividades. O Moodle permite uma customização interna, manual e dinâmica por usuário. Assim, o professor pode adicionar blocos (calendário, eventos, etc.) à interface.

Neste ambiente é possível uma customização para atender diferentes instituições, departamentos e cursos. O Moodle, também, permite uma customização interna, automática e dinâmica por usuário, pois o ambiente modifica-se automaticamente, adaptando-se de acordo com cada usuário do ambiente de maneira dinâmica.

#### 5.2.6. Conceito “portabilidade para dispositivos” no Moodle

No mapa geral, a portabilidade para dispositivos indica se o ambiente oferece suporte para ser acessado a partir de diferentes dispositivos. No entanto, no Moodle não oferece suporte a TV Digital.

#### 5.2.7. Componente "internacionalização” no Moodle

Este componente, no mapa geral, refere-se à capacidade do ambiente de oferecer suporte para que seus cursos possam ser acessados em diferentes idiomas, possua formato

para diferentes moedas e data/hora. O Moodle oferece suporte a diferentes idiomas, mas a tradução não é total, parte do ambiente fica em inglês. Oferece suporte também à data e hora.

#### 5.2.8. Conceito “serviços” no Moodle

No mapa geral, o conceito serviços, trata dos recursos utilizados no desenvolvimento do curso, podendo ser classificados como, serviços internos ou externos. E esses ainda podem ser serviços administrativos, didáticos, de avaliação, de comunicação, individuais, de grupo, de inteligência.

Sendo assim, o Moodle possui serviços para todas as categorias dos serviços internos. Não foi encontrado nenhum serviço externo. Para os serviços internos administrativos, os serviços que permitem que professor, possa gerenciar o seu curso, por exemplo, pastas, rótulos, páginas, upload de arquivos, backup do curso, restauração do curso, agenda do curso, fórum de notícias, últimas notícias, próximos eventos, divulgação de notas e relatórios.

Nos internos didáticos, o Moodle implementa os serviços interativo, como textos online (construção de texto pelo aluno com comentários do professor), fórum, chat, Workshop e Wiki. Além de, também, implementar os não interativos, como entrega de material e referências na web – links, entre outros como transparências, material de referência do curso, documentos multimídia, glossário indexado, base de dados (permitindo inserir documentação) e questionário.

Para avaliar os alunos os professores encontram na Moodle, provas, exercícios, tarefas, fóruns (há a possibilidade de atribuir nota). Para os serviços internos de comunicação o Moodle implementa tanto serviços síncronos quanto assíncronos. Como serviço síncrono há o chat e como assíncrono há agenda do curso, fóruns, blog, próximos eventos e últimas notícias. Para os serviços internos de grupo há formação de grupo de discussão, compartilhamento de materiais. Nos serviços individuais há os perfis dos usuários e os portfólios privado, onde os usuários colocam materiais que somente ele tem acesso. No caso do professor, materiais que ele ainda vai passar para os alunos.

#### 5.2.9. Conceito “grupos” no Moodle

Este conceito, no mapa geral, trata da possibilidade de formação e organização de grupos entre os usuários do ambiente dentro do curso. O Moodle permite que o professor da disciplina crie grupos facilmente e determine como os membros se relacionam com os outros grupos e nas diferentes atividades. No mapa geral, são considerados quatro aspectos:

i) Definição do grupo que pode ser simples, sendo espontâneo ou moderado, ou hierárquico, também espontâneo ou moderado; Quando obedecem a uma hierarquia, os grupos podem ser classificados como: grupo geral, grupos isolados e conjunto de grupos. No Moodle, os grupos podem ser definidos como simples ou hierárquico, mas todos são moderados, ou seja, criados por o professor ou por o administrador.

ii) Controle de visibilidade é uma definição de visibilidade entre os membros do grupo, pode ser fixa ou configurável. Não existe controle de visibilidade entre os grupos do Moodle;

iii) Interação é a possibilidade de comunicação, cooperação e colaboração entre grupos. Pode ser inter-grupos e intra-grupos. No Moodle é possível haver comunicação inter-grupos;

iv) Grupos Externos é a possibilidade de integração com grupos da web externos ao ambiente. Este aspecto, também, não foi encontrado no Moodle.

### **5.3. Claroline**

O Claroline é um ambiente virtual de aprendizagem que foi iniciado em 2001 pela Universidade Católica de Louvain na Bélgica, sendo financiado pela Fundação de Louvain. Seus principais desenvolvedores são Thomas De Praetere e Marcel Lebrun. Foi desenvolvido por professores e está disponível em português.

Desde 2004, o código do Claroline (2011) é desenvolvido em parceria com o CERDECAM8, centro de pesquisas da ECAM9 (Escola de Engenharia da Bélgica). Está disponível para download em <http://www.claroline.net>. O software, voltado para a Web, foi desenvolvido utilizando a linguagem PHP e MySQL para gerenciamento de dados. O mapa conceitual deste ambiente é representado nas Figuras 23 e 24.

Figura 22 - Mapa Conceitual do Ambiente Claroline – parte 1

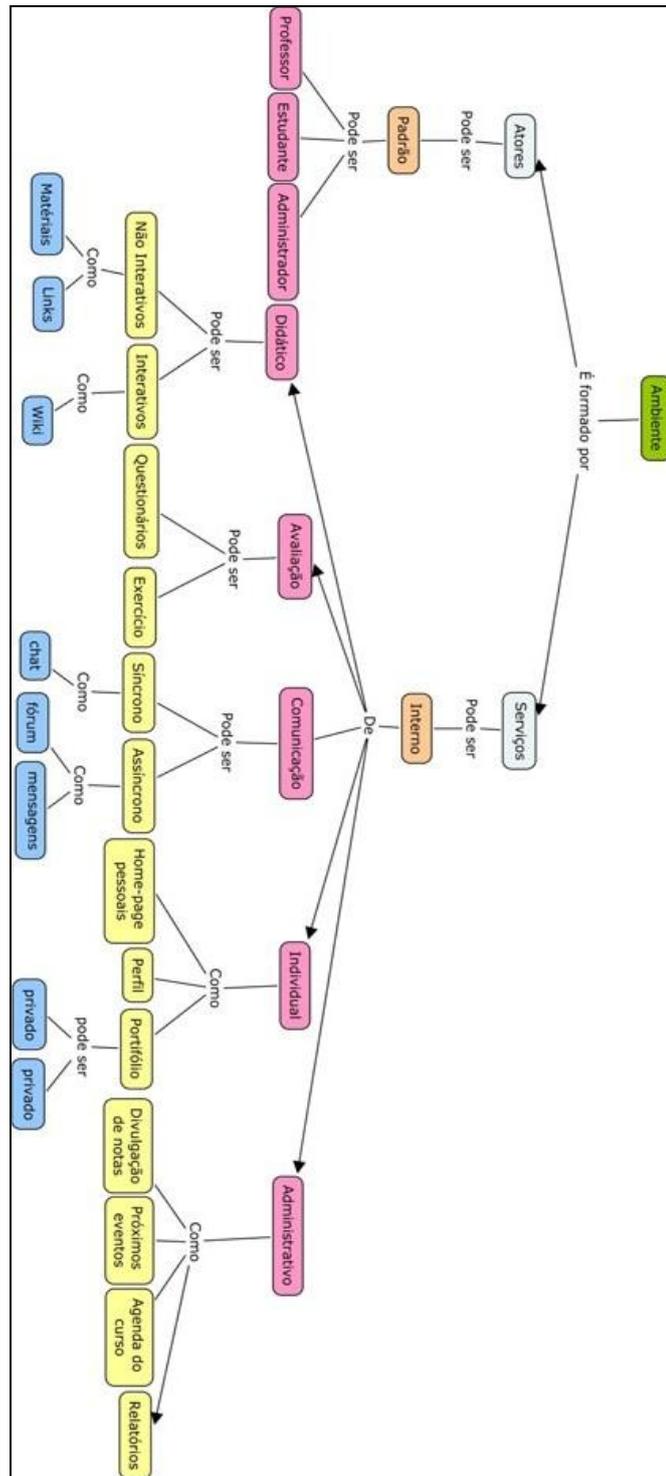
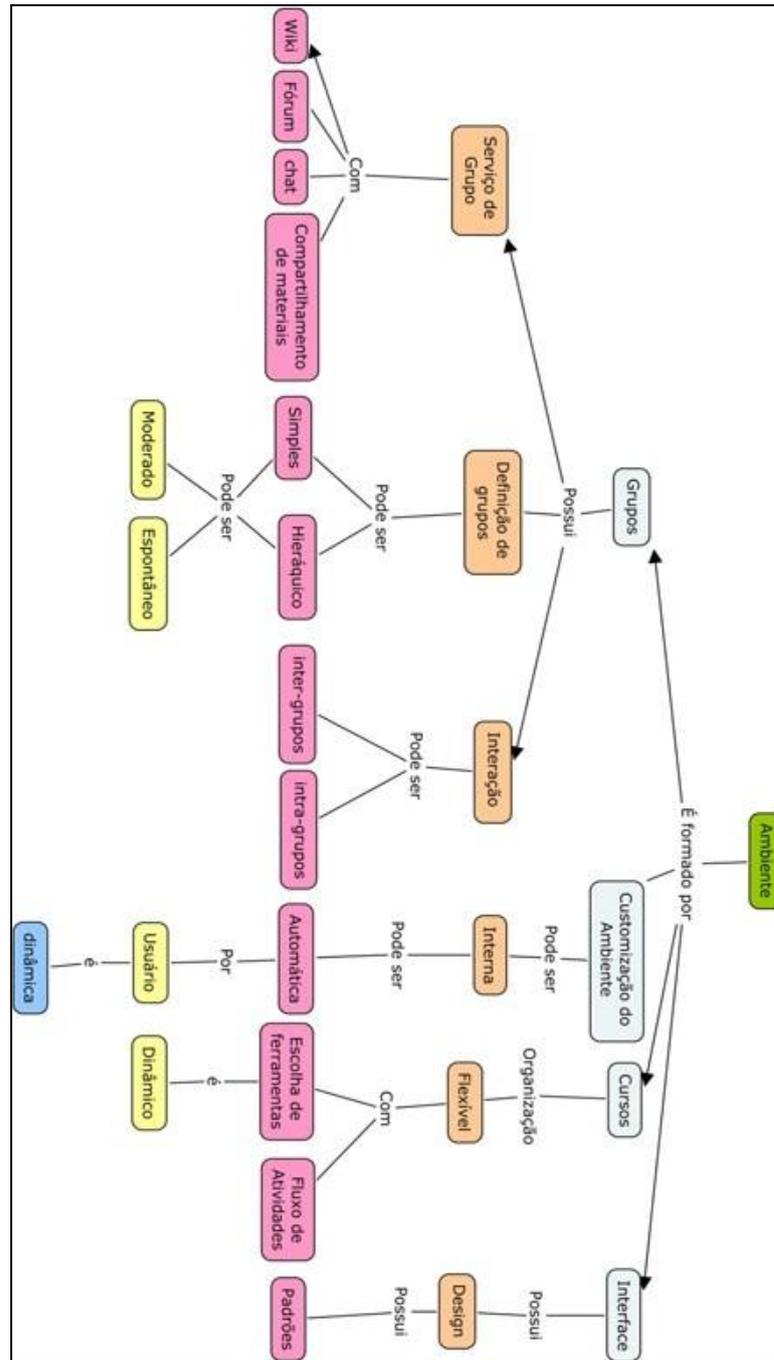


Figura 23 - Mapa Conceitual do Ambiente Claroline – parte 2



### 5.3.1. Conceito “atores” no Claroline

O conceito “atores”, no mapa geral, considera a existência de atores que interagem com o ambiente. Esses atores podem existir de duas formas: fixa ou flexível.

O Claroline trabalha, de forma fixa, com 3 atores, cada um com suas especificidades no sistema. São eles:

- Estudante: Este é um usuário que pode acessar apenas os materiais e ferramentas dos cursos criados por um professor.

- Professor: Esta classe de usuário tem todos os direitos dos estudantes e também tem o direito de criar novos cursos e modificar todo o seu conteúdo.
- Administrador: É o tipo de usuário que tem os direitos mais extensos na plataforma. Pode-se atribuir direitos especiais a todas as classes de usuários e é gravado normalmente para os professores.

### 5.3.2. Conceito “interface” no Claroline

No conceito “interface”, o Claroline deixou a desejar em vários subconceitos. No conceito usabilidade, que considera a qualidade de interação com os usuários, ele não se mostrou fácil de aprender, pois os caminhos são confusos e parecidos e os ícones não são sugestivos.

O ambiente também não é fácil usar, pois requer um pouco esforço. Não é um ambiente agradável de trabalhar. Por não ser fácil de aprender e não ser fácil de usar torna-se improdutivo. No design, ele apresenta certa harmonia entre cores utilizadas, além dos tipos e formatos de letras e caixas de diálogos que se mantêm ao longo de todo o ambiente.

### 5.3.3. Conceito “cursos” no Claroline

No mapa geral, este conceito busca analisar o suporte que o ambiente oferece para a estrutura e organização de cursos. Para estruturar e organizar os cursos, no Claroline, existe um espaço reservado para o curso, onde neste mesmo espaço, há um espaço para cada serviço. Assim, o professor cria e alunos consultam os serviços do curso em cada espaço desses. A organização e estruturação dos cursos, no mapa geral, podem ocorrer das seguintes formas:

- a) Padrão, quando ela é fixa para todo e qualquer curso, no entanto, o Claroline não implementa esse tipo de organização e estruturação.
- b) Flexível, quando na estrutura do curso existe a flexibilidade tanto de escolher os serviços que serão utilizados no curso, como também o fluxo das atividades. No Claroline, apesar de ser em partes padrão, por ter áreas fixas para colocar cada tipo serviço, ele possui flexibilidade na escolha de serviços. No mapa geral, a flexibilidade de cursos é composta pelos conceitos:
  - i. Escolha de serviços: o professor tem autonomia para determinar, por exemplo, se vai utilizar um fórum ou chat no curso, essa escolha pode ser feita de duas formas: estática ou dinâmica. No Claroline, a escolha de serviços é feita de forma dinâmica, o usuário professor pode realizar modificações, ou seja, adicionar e remover

serviços, a qualquer momento do curso;

- ii. Fluxo de atividades: Cada professor pode escolher a sequência das ferramentas que serão trabalhadas. O Claroline não oferece fluxo para as atividades.

#### 5.3.4. Conceito “percepção do ambiente” no Claroline

No mapa geral, este conceito, são tratados aspectos de percepção de tarefas e atividades, individuais e do grupo, como também a percepção social dentro do ambiente. Sendo subdividido em: percepção social e percepção de tarefas.

Para a percepção social, no Claroline, o usuário pode visualizar que participa do curso que o mesmo, também, participa na opção “Utilizadores”. Também é possível visualizar os participantes dos grupos de cada curso.

Quanto à percepção de tarefas, o Claroline oferece na tela inicial dois serviços um que mostra a atividade em si e outro que mostra um calendário com a data final para o envio da atividade.

#### 5.3.5. Conceito “customização do ambiente” no Claroline

No mapa geral, este conceito abrange todas as mudanças que podem ocorrer na organização, conteúdo e interface do ambiente. Essa customização pode ser especificada como, interna com adaptação manual ou automática para usuário, de forma estática ou dinâmica e adaptação manual ou automática para instituições e departamentos; ou externa, com a integração manual ou automática de ferramentas e AVA.

O Claroline não possui boa flexibilidade, pois apenas oferece pequenas alterações no ambiente de um tipo de usuário para outro. Exemplo: professor para estudante.

Sendo assim, o Claroline, permite uma customização interna, automática e dinâmica por usuário. Dependendo do tipo de usuário, diferentes funcionalidades e conteúdos serão disponibilizados. Por a customização ser dinâmica, o ambiente pode sofrer alterações significativas o longo do curso.

#### 5.3.6. Conceito “portabilidade para dispositivos” no Claroline

No mapa geral, a portabilidade para dispositivos indica se o ambiente oferece suporte para ser acessado a partir de diferentes dispositivos. No entanto, no Claroline não oferece suporte a TV Digital.

### 5.3.7. Conceito “internacionalização” no Claroline

Este conceito, no mapa geral, refere-se à capacidade do ambiente de oferecer suporte para que seus cursos possam ser acessados em diferentes idiomas, possua formato para diferentes moedas e data/hora. O Claroline oferece suporte a diferentes idiomas como, inglês, francês, espanhol, português. Ele foi traduzido para 35 idiomas. Oferece suporte também à data e hora.

### 5.3.8. Conceito “serviços” no Claroline

No mapa geral, o conceito serviços, trata dos recursos utilizados no desenvolvimento do curso, podendo ser classificados como, serviços internos ou externos. E esses ainda podem ser serviços administrativos, didáticos, de avaliação, de comunicação, individuais, de grupo, de inteligência.

Sendo assim, o Claroline, com exceção dos serviços internos inteligentes, possui serviços para as demais categorias de serviços internos. Não foi encontrado nenhum serviço externo. Para os serviços internos administrativos, os serviços que permitem que professor possa gerenciar o seu curso, o Claroline possui agenda de tarefas e prazos, estatísticas de cursos, com informações como: siga o acesso à plataforma, siga as ferramentas utilizadas, monitorar a progressão de usuários, relatórios e pastas.

Nos serviços internos didáticos, o Claroline disponibiliza os serviços interativos, como construção de documentos de forma colaborativa, fórum, debates, Workshop e Wiki. Além de, também, implementar os não interativos como tarefas, exercícios e documentos.

Como serviços de avaliação, o Claroline oferece exercícios, tarefas, fóruns. Para os serviços internos de comunicação, são implementados tanto serviços síncronos quanto assíncronos. Como serviço síncrono há o chat e como assíncrono há agenda, fóruns, wikis, anúncios e etc. Para os serviços internos de grupo, no Claroline, todos serviços que são disponíveis para cursos, também são disponibilizados de forma restrita para cada grupo. Nos serviços individuais há apenas os perfis dos usuários.

### 5.3.9. Conceito “grupos” no Claroline

Este conceito, no mapa geral, trata da possibilidade de formação e organização de grupos entre os usuários do ambiente dentro do curso. O Moodle permite que o professor da disciplina crie grupos facilmente e determine como os membros se relacionam com os outros grupos e nas diferentes atividades. No mapa geral, são considerados quatro aspectos:

i) Definição do grupo que pode ser simples, sendo espontâneo ou moderado, ou hierárquico, também espontâneo ou moderado; Quando obedecem a uma hierarquia, os grupos podem ser classificados como: grupo geral, grupos isolados e conjunto de grupos. No Claroline, os grupos podem ser definidos como simples ou hierárquico, mas todos são moderados, ou seja, criados por o professor ou por o administrador. Algo que merece destaque na criação de grupos para este ambiente é o serviço portfólio de grupos, pois é um espaço reservado dentro ambiente para postagem de materiais referentes apenas a um determinado grupo;

ii) Controle de visibilidade é uma definição de visibilidade entre os membros do grupo, pode ser fixa ou configurável. Não existe controle de visibilidade entre os grupos do Claroline;

iii) Interação é a possibilidade de comunicação, cooperação e colaboração entre grupos. Pode ser inter-grupos e intra-grupos. No Claroline não existe nenhum meio de comunicação inter-grupos, mas há comunicação intra-grupos;

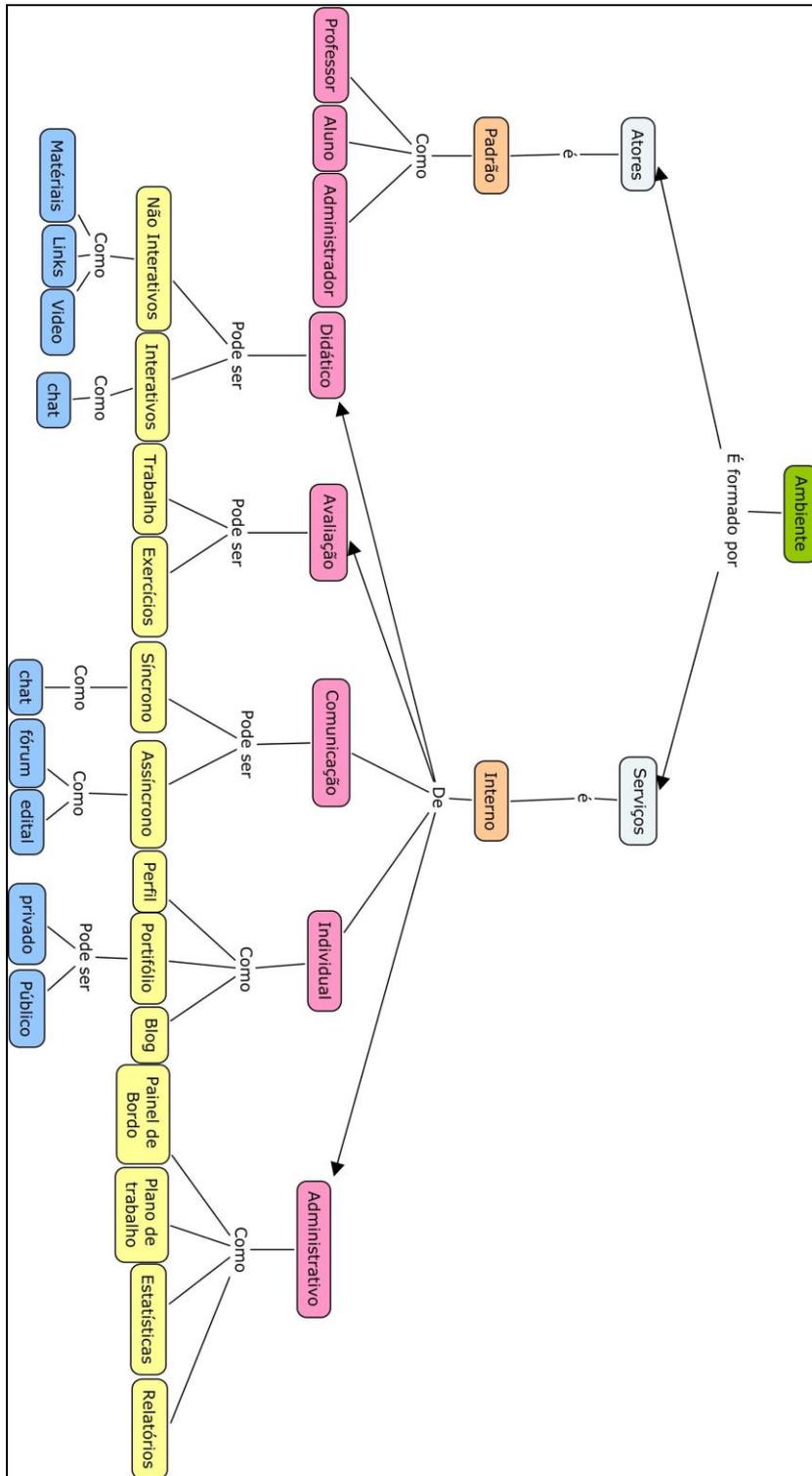
iv) Grupos Externos é a possibilidade de integração com grupos da web externos ao ambiente. Este aspecto, também, não foi encontrado no Claroline.

#### **5.4. Eureka**

O Eureka (2011) foi desenvolvido pelo Laboratório de Mídias Interativas (LAMI) da Pontifícia Universidade Católica do Paraná (PUCPR) em 1999.

“O ambiente foi utilizado tanto pela Siemens, em cursos de treinamento profissional a distância, quanto pela PUCPR, como apoio aos cursos de graduação, pós-graduação presenciais e para pesquisa.” (Ferreira, 2007) “Seu principal diferencial em relação às plataformas observadas é a utilização de áudio do texto escrito em todas as telas acessadas. A qualidade da locução é bastante nítida, inclusive em relação ao sotaque característico da região” (Gabardo, 2010). O mapa conceitual para este ambiente pode ser observado nas Figuras 25 e 26.

Figura 24 -Mapa Conceitual do Ambiente Eureka - parte 1





Este conceito é considerado um dos pontos fortes do Eureka, pois este ambiente oferece uma acessibilidade parcial para deficientes visuais. Através de um leitor de tela é possível ouvir o que está sendo apresentado na tela. Essa interface é dividida em três regiões: à esquerda e à direita estão caixas com ferramentas de acesso geral, que afetam todo o ambiente como um todo, e a região central. Para o conceito “interface”, o Eureka mostrou-se bem em vários subconceitos. No conceito usabilidade, que considera a qualidade de interação com os usuários, ele se mostrou fácil de aprender, pois os caminhos são curtos, os ícones são sugestivos e a interface é leve.

O ambiente também é fácil usar, pois não requer um grande esforço, além de oferecer um “tour” para o visitante torna o ambiente aberto a visitantes. É um ambiente agradável de trabalhar. No entanto não é produtivo, pois para realizar algumas atividades o aluno precisa utilizar ferramenta que não são oferecidas pelo ambiente e depois postá-la no mesmo. No design, o layout se destaca pelo uso de ilustrações e informações bem distribuídas, em três ambientes, com predomínio do azul em fundo branco.

#### 5.4.3. Conceito “cursos” no Eureka

Existe uma flexibilidade para o usuário acessar as atividades. Uma vez escolhido o perfil, somente as salas e os conteúdos referentes a esse perfil estarão disponíveis. Existem várias maneiras de acessar os conteúdos, depende de como são organizadas as salas e como são inseridos os dados. Podem-se acessar as atividades a partir do início, pelas salas ou pela agenda, ou a partir do item “salas”, nesse caso terá que selecionar a sala pretendida.

No Eureka, os cursos são chamados de “salas”, estas salas são categorizadas em: Ativas, Encerradas e Cancelas. Nas salas existem os nomes e navegações das salas, que são, por exemplo, nome do curso, ano e semestre. Dentro da sala existe um menu como as opções: arquivos, grupos, comunicação, estudos, painel de bordo e sair da sala.

A agenda permite visualizar as atividades e acessá-las. Passando o mouse por cima dos campos aparecem as informações sobre a atividade. Exemplo, o índice 4/5 significa que é o dia 4 dos 5 dias estipulados da tarefa, falta um dia para que o prazo da realização seja esgotado.

Este conceito busca analisar o suporte que o ambiente oferece para que o professor possa estruturar seus cursos. O professor tem liberdade para estrutura o curso da maneira que desejar. A organização e estruturação dos cursos, no Eureka, são da seguinte forma:

- i) A escolha de serviços no Eureka de forma dinâmica, o usuário professor pode realizar

- modificações, ou seja, adicionar e remover serviços, a qualquer momento do curso;
- ii) No Eureka, o professor pode determinar fluxo de atividades. Cada usuário professor pode escolher a sequência das ferramentas que serão trabalhadas. As atividades estão distribuídas em um calendário, no qual se pode ver a data de início, a de fim e o decorrer da atividade no tempo. Também é possível visualizar o número de dias destinados a cada atividade e em que dia do total você se encontra.

#### 5.4.4. Conceito “percepção do ambiente” no Eureka

No mapa geral, este conceito, são tratados aspectos de percepção de tarefas e atividades, individuais e do grupo, como também a percepção social dentro do ambiente. Sendo subdividido em: percepção social e percepção de tarefas.

Neste conceito, o Eureka, para a percepção social, oferece um recurso, Usuários Online, que lista os usuários que estão acessando o ambiente naquele momento, Oferece outro recurso também, no item participantes o professor tem uma lista rápida de todos os participantes (tutores e alunos) do curso, sendo possível também consultar os seus perfis.

Na percepção de tarefas, o Eureka oferece um recurso que indica as atividades recentes no ambiente e outro que o professor pode visualizar os alunos que estão enviando as atividades. O serviço “quadro de avisos” disponibiliza uma relação de eventos em ordem cronológica informando inclusive a hora para seu término. Compreende-se por evento qualquer atividade com prazo definido, como: entrega de trabalho, fechamento de fóruns, inscrição em disciplinas, etc.

#### 5.4.5. Conceito “customização do ambiente” no Eureka

No mapa geral, este conceito abrange todas as mudanças que podem ocorrer na organização, conteúdo e interface do ambiente. Essa customização pode ser especificada como, interna com adaptação manual ou automática para usuário, de forma estática ou dinâmica e adaptação manual ou automática para instituições e departamentos; ou externa, com a integração manual ou automática de ferramentas e AVA.

Os recursos do Eureka possuem boa flexibilidade e permitem muitas formas de configurações. Por ser de código aberto, o Eureka permite a integração com outras ferramentas de forma manual proporciona a integração com “plugins” que podem ser desenvolvidos como recursos ou atividades.

O Eureka permite uma customização interna, manual e dinâmica por usuário. Assim, o professor pode adicionar blocos (calendário, eventos e etc.) à interface. O Eureka, também, permite uma customização interna, automática e dinâmica por usuário. Em função do perfil escolhido, diferentes funcionalidades e conteúdos serão disponibilizados.

#### 5.4.6. Conceito “portabilidade para dispositivos” no Eureka

No mapa geral, a portabilidade para dispositivos indica se o ambiente oferece suporte para ser acessado a partir de diferentes dispositivos. No entanto, no Eureka não oferece suporte a TV Digital.

#### 5.4.7. Conceito "internacionalização” no Eureka

Este conceito, no mapa geral, refere-se à capacidade do ambiente de oferecer suporte para que seus cursos possam ser acessados em diferentes idiomas, possua formato para diferentes moedas e data/hora. O mesmo curso no Eureka não pode ser acessado em diferentes idiomas.

#### 5.4.8. Conceito “serviços” no Eureka

No mapa geral, o conceito serviços, trata dos recursos utilizados no desenvolvimento do curso, podendo ser classificados como, serviços internos ou externos. E esses ainda podem ser serviços administrativos, didáticos, de avaliação, de comunicação, individuais, de grupo, de inteligência.

Assim, o Eureka possui serviços para praticamente todas as categorias dos serviços internos, menos os de inteligência. Não foi encontrado nenhum serviço externo. Para os serviços internos administrativos, os serviços que permitem que professor, possa gerenciar o seu curso, por exemplo, plano de trabalho, painel de bordo, agenda de provas, estatísticas, relatórios e dados da sala.

Para os serviços internos didáticos interativos, o Eureka implementa apenas o chat. Além de, também, implementar os não interativos, material didático on-line e webgrafia,.

Para avaliar os alunos os professores encontram no Eureka, exercícios, tarefas, fóruns. Nos serviços internos de comunicação o Eureka implementa tanto serviços síncronos quanto assíncronos. Como serviço síncrono há apenas o chat e como assíncrono há edital, fóruns e aviso do sistema. Para os serviços internos de grupo há formação de grupo de discussão, compartilhamento de materiais. Nos serviços individuais há o portfólio que está dividido em

particular: acessível somente a ele, com uma pasta pessoal de um limite total de 3Mb; público: onde o usuário pode também copiar arquivos ou pastas disponíveis no sua pasta particular para a área “arquivos” de uma sala específica do EUREKA ou vice-versa. Basta clicar sobre o botão “copiar” quando acessar os dados do arquivo ou da pasta e selecionar a sala de destino para o arquivo que usuário deseja copiar.

#### 5.4.9. Conceito “grupos” no Eureka

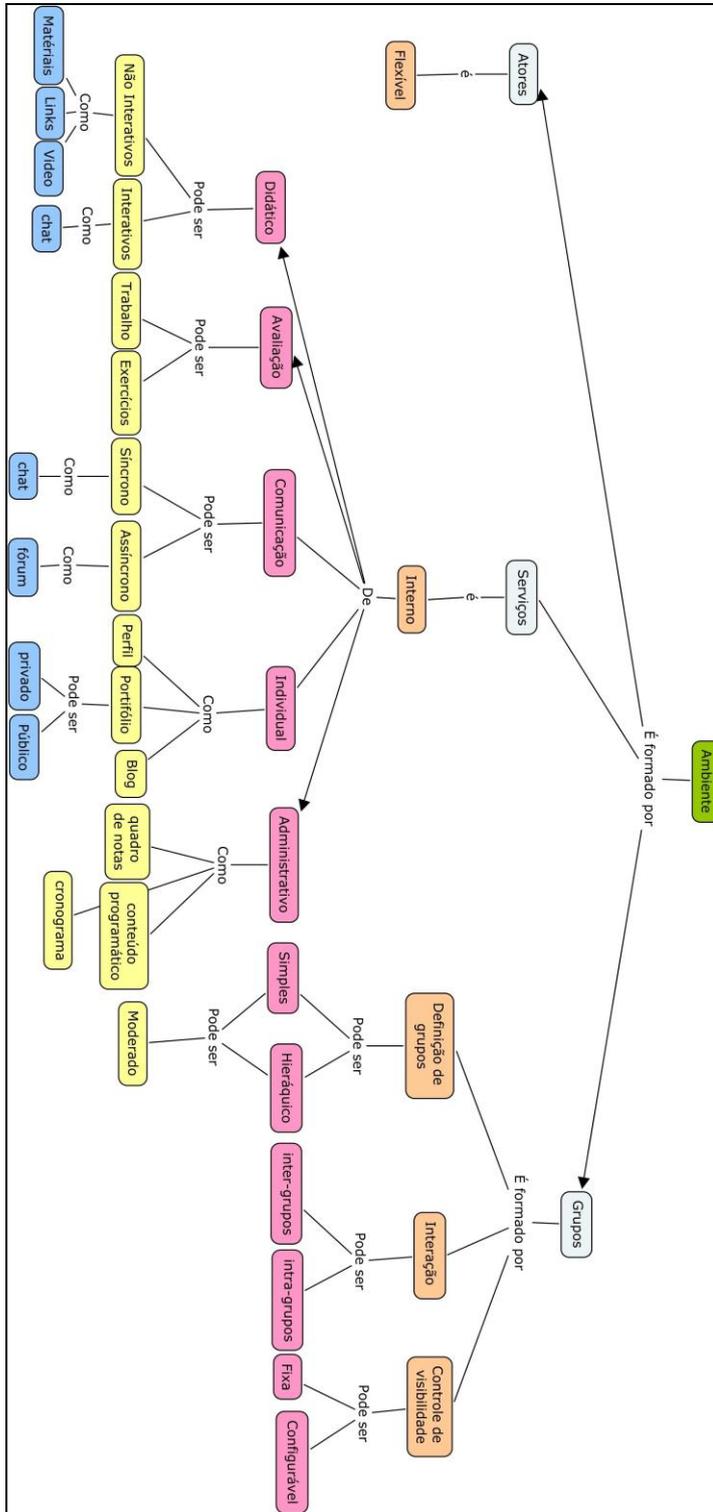
Este conceito, no mapa geral, trata da possibilidade de formação e organização de grupos entre os usuários do ambiente dentro do curso. O Eureka não oferece suporte a esse conceito.

### 5.5. Sakai

O *Sakai* é um AVA, iniciado em 2004. Financiado por uma doação da Fundação Mellon, o *Sakai* foi construído como uma colaboração entre Stanford, Indiana, Michigan, MIT e Berkeley, para ser utilizado no ensino superior (*SAKAI*). Foi baseado em ferramentas existentes em cada uma das instituições fundadoras.

*Sakai* foi lançado ao público em 2005 e é gerenciado hoje pela Fundação *Sakai*, que supervisiona o seu desenvolvimento e roteiro do projeto. O AVA é programado em Java e projetado para ser uma suíte de aplicativos orientada a serviços. Alcançou uma boa reputação por utilizar recursos de ponta, escalabilidade e segurança, e ficou popular em grandes universidades que precisam de uma solução robusta.

Figura 26 - Mapa Conceitual do Ambiente Sakai - parte 1





ambiente possui uma interface estruturada. A organização da parte superior da interface é fixa, assim apenas o conteúdo do meio é alterado sendo alterado apenas o meio, tornando o ambiente fácil de aprender e aumentando a produtividade. Além disso, a forma de apresentação do curso é agradável. No design, o Sakai apresentou um o layout bem organizado e estruturado.

### 5.5.3. Conceito “cursos” no Sakai

Este conceito busca analisar o suporte que o ambiente oferece para que o professor possa organizar e estruturar seus cursos. Os cursos do Sakai são na forma de sites, onde cada um tem seu portfólio. A organização e estruturação dos cursos, no Sakai, são da seguinte forma:

- i) No Sakai, o professor pode determinar fluxo de atividades de forma dinâmica, o usuário professor pode realizar modificações, ou seja, adicionar e remover serviços, a qualquer momento do curso;
- ii) No Sakai, o professor pode determinar fluxo de atividades. Cada usuário professor pode escolher a sequência das ferramentas que serão trabalhadas, atribuindo data de início e fim de uma atividade.

### 5.5.4. Conceito “percepção do ambiente” no Sakai

No mapa geral, neste conceito, são tratados aspectos de percepção de tarefas e atividades, individuais e do grupo, como também a percepção social dentro do ambiente. Sendo subdividido em: percepção social e percepção de tarefas.

Neste conceito, o Sakai, para a percepção social, oferece o recurso “Participantes”, que fica localizado na página de cada curso. Assim, o usuário professor ou o usuário aluno pode visualizar os integrantes de cada curso. Na percepção de tarefas, o Sakai possui o recurso “Announcements”, nele o usuário pode consultar todo tipo de avisos destinados a ele.

### 5.5.5. Conceito “customização do ambiente” no Sakai

No mapa geral, este conceito abrange todas as mudanças que podem ocorrer na organização, conteúdo e interface do ambiente. Essa customização pode ser especificada como, interna com adaptação manual ou automática para usuário, de forma estática ou dinâmica e adaptação manual ou automática para instituições e departamentos ou externa, com a integração manual ou automática de ferramentas e AVA.

Por ser de código aberto, o Sakai permite a integração com outras ferramentas de forma manual e automática, proporcionando a integração com *plugins* que podem ser desenvolvidos como recursos ou atividades. O Sakai permite uma customização interna, manual e dinâmica por usuário.

Neste ambiente é possível uma customização para atender diferentes instituições, departamentos e cursos. O Sakai, também, permite uma customização interna, automática e dinâmica por usuário, pois o ambiente modifica-se automaticamente, adaptando-se de acordo com cada usuário do ambiente de maneira dinâmica.

#### 5.5.6. O Conceito “portabilidade para dispositivos” no Sakai

No mapa geral, a portabilidade para dispositivos indica se o ambiente oferece suporte para ser acessado a partir de diferentes dispositivos. No entanto, no Sakai não foi encontrado suporte para TV Digital.

#### 5.5.7. Conceito “internacionalização” no Sakai

Este conceito, no mapa geral, refere-se à capacidade do ambiente de oferecer suporte para que seus cursos possam ser acessados em diferentes idiomas, possua formato para diferentes moedas e data/hora. O Sakai oferece suporte para que seus cursos sejam acessados em diferentes idiomas, moedas e datas.

#### 5.5.8. Conceito “serviços” no Sakai

No mapa geral, o conceito serviços, trata dos recursos utilizados no desenvolvimento do curso, podendo ser classificados como, serviços internos ou externos. E esses ainda podem ser serviços administrativos, didáticos, de avaliação, de comunicação, individuais, de grupo, de inteligência.

O Sakai possui serviços para todas as categorias dos serviços internos, menos os de inteligência. Para os serviços internos administrativos, os serviços que permitem que professor, possa gerenciar o seu curso, por exemplo, escaninho (usado na troca de documentos entre alunos e professores), quadro de notas, cronograma e conteúdo programático.

Para os serviços didáticos interativos, o Tidia-ae implementa o serviço “chat”, e o “comunicador instantâneo”. Além de, também, implementar os não interativos, material

didático on-line e webgrafia. Para avaliar os alunos os professores encontram no Sakai, exercícios, tarefas, fóruns.

Nos serviços internos de comunicação o Tidia-ae implementa os serviços assíncronos, por exemplo, o serviço “Mensagens” que permite enviar e receber mensagens e “Discussão” um tipo fórum. Além de implementar uma diversidade de serviços síncronos, por exemplo, o serviço “bate-papo”, “whiteboard” e o “comunicador instantâneo”. Nos serviços individuais há o portfólio, que está dividido em particular: acessível somente a ele e público: compartilhado com todos os usuários do ambiente. Além do “profile” espaço no qual os usuários podem publicar informações pessoais, incluindo foto. No Tidia-ae o usuário possui o “meu site”, onde o usuário pode verificar seus compromissos, armazenar seus próprios documentos e criar outros sites.

#### 5.5.9. Conceito “grupos” no Sakai

Este conceito, no mapa geral, trata da possibilidade de formação e organização de grupos entre os usuários do ambiente dentro do curso. O Sakai permite que o professor da disciplina crie grupos facilmente e determine como os membros se relacionam com os outros grupos e nas diferentes atividades. No mapa geral, são considerados quatro aspectos:

i) Definição do grupo que pode ser simples, sendo espontâneo ou moderado, ou hierárquico, também espontâneo ou moderado; Quando obedecem a uma hierarquia, os grupos podem ser classificados como: grupo geral, grupos isolados e conjunto de grupos. No Sakai, os grupos podem ser definidos como simples ou hierárquico, mas todos são moderados, ou seja, criados por o professor ou por o administrador.

ii) Controle de visibilidade é uma definição de visibilidade entre os membros do grupo, pode ser fixa ou configurável. Existe este controle de visibilidade entre os grupos Sakai;

iii) Interação é a possibilidade de comunicação, cooperação e colaboração entre grupos. Pode ser inter-grupos e intra-grupos. No Sakai meios de comunicação inter-grupos e comunicação intra-grupos;

iv) Grupos Externos é a possibilidade de integração com grupos da web externos ao ambiente. Este aspecto, não foi encontrado no Sakai.

## 5.6. Tidia-ae

De acordo com Oliveira (2009) o TIDIA-AE é um LMS desenvolvido pela Universidade de São Paulo. Ele possui ferramentas que dão suporte ao ensino presencial e a distância. Segundo Projeto Tidia – AE (2004, citado por Franciscato, 2008) “o AVA Tidia - Ae é um ambiente colaborativo que gerencia cursos e atividades de aprendizado, dando suporte ao ensino presencial e a distância”. Vale ressaltar que este ambiente foi desenvolvido utilizando o mesmo framework que o ambiente anterior, o *Sakai*.

No tidia-ae tudo referente a um usuário está em uma página do ambiente chamada “meu site”, nela fica não só o portfólio, mas também verificação compromissos e criação novos sites. O mapa conceitual representando este ambiente pode ser observado nas Figuras 29 e 30.

Figura 28 - Mapa Conceitual do Ambiente Tidia Ae - parte 1

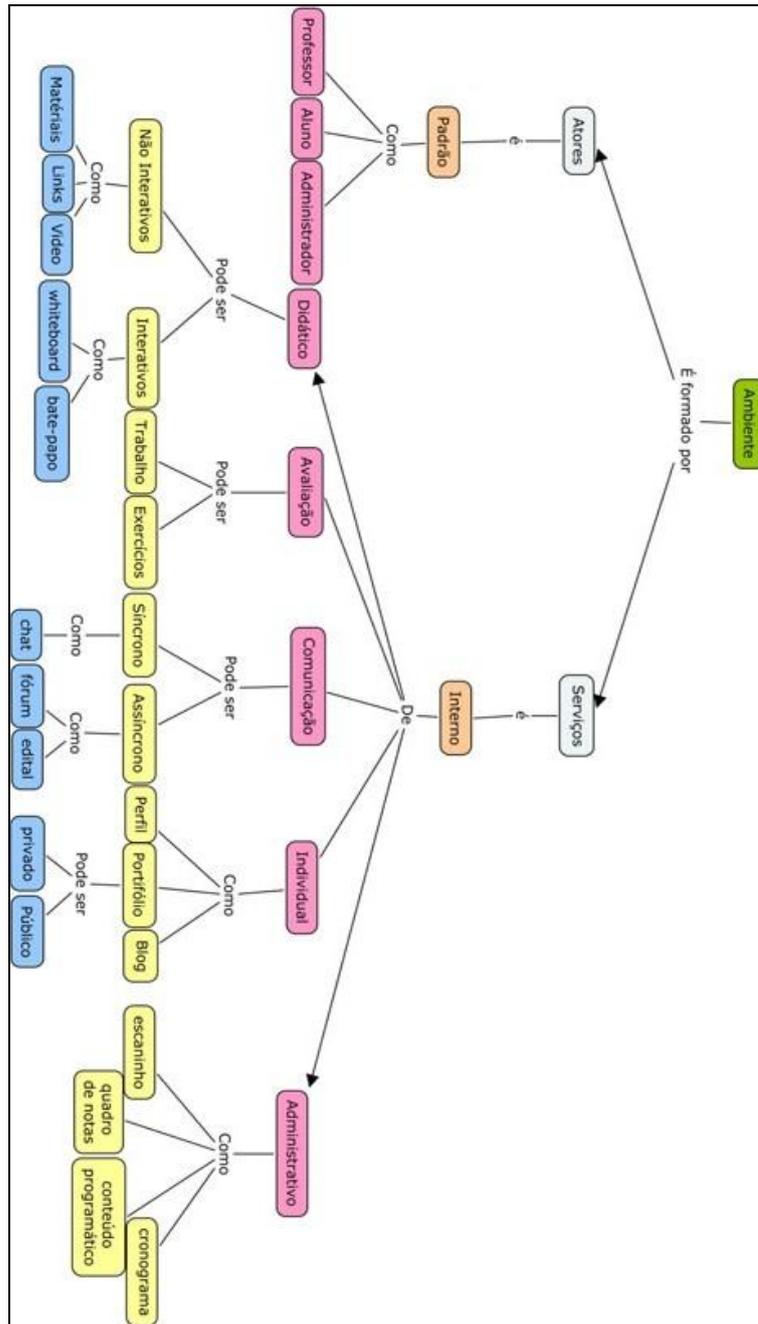
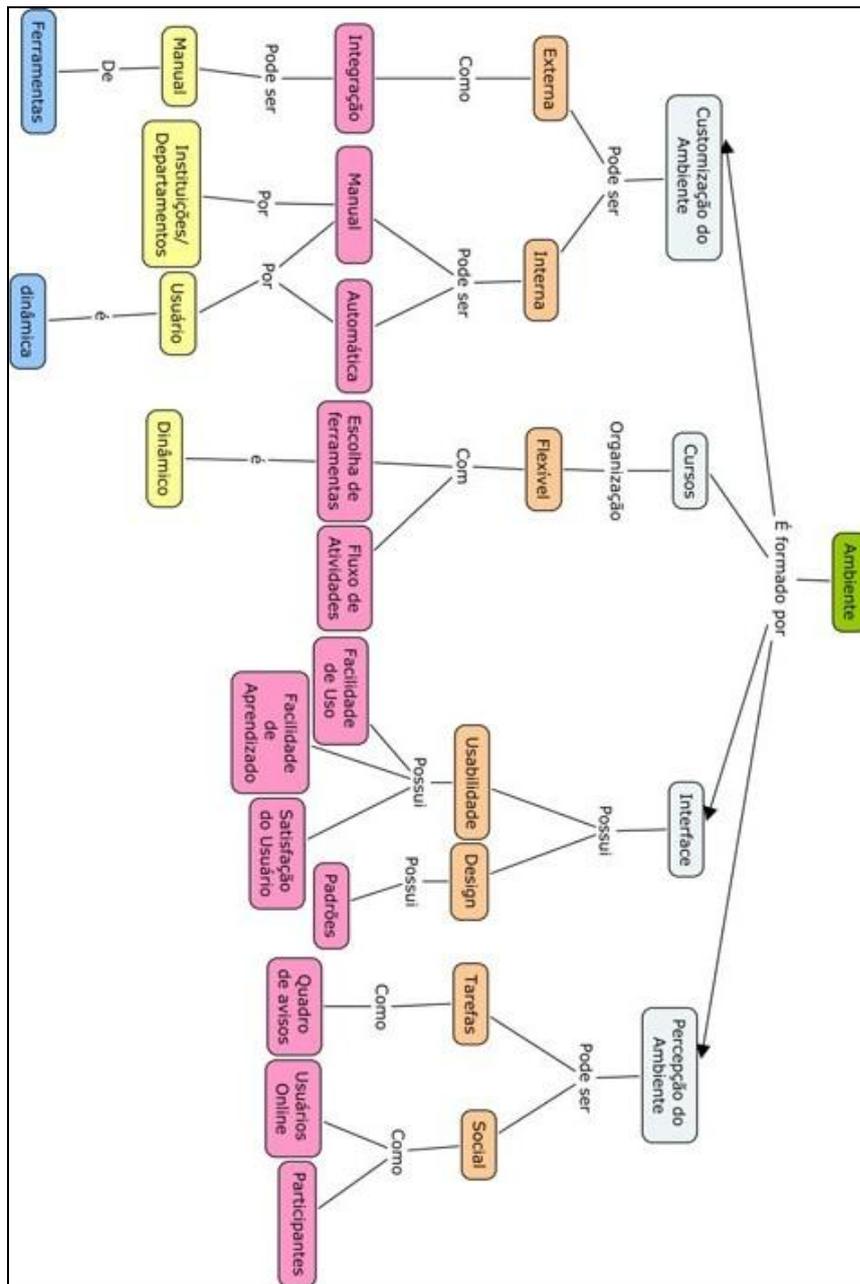


Figura 29 - Mapa Conceitual do Ambiente Tidia Ae - parte 2



### 5.6.1. Conceito “atores” no Tidia-ae

Este conceito, no mapa geral considera a existência de atores que interagem com o ambiente. Esses atores podem existir de duas formas: fixa ou flexível. O Tidia-ae não permite flexibilidade para criação dos atores. No entanto, este ambiente possui, basicamente, dois perfis fixos: aluno e professor.

### 5.6.2. Conceito “interface” no Tidia-ae

Para este conceito o Tidia-ae, apresentou o desempenho razoável, pois no subconceito usabilidade, que considera a qualidade de interação com os usuários, este ambiente possui uma interface organizada e que se altera pouco na navegação. A organização da parte superior e as laterais da interface são fixas, sendo alterado apenas o meio. Assim, torna-se fácil de usar e de aprender, além de aumentar a produtividade. No entanto, a forma como é apresentado o conteúdo do curso, quando comparado com outros ambientes, é um pouco desagradável.

No design, o Tidia-ae apresentou um layout bem organizado, em três partes, com um menu lateral, o topo com as abas dos cursos e o centro que sofre mais alteração durante a interação, no entanto, nesta parte a forma como é apresentado o conteúdo se mostra diferente das demais, pois são utilizados estilos de formatação diferentes.

### 5.6.3. Conceito “cursos” no Tidia-ae

Este conceito busca analisar o suporte que o ambiente oferece para que o professor possa organizar e estruturar seus cursos. Os cursos do Tidia-ae são na forma de sites, onde cada um tem seu portfólio. A organização e estruturação dos cursos, no Tidia-ae, são da seguinte forma:

- iii) No Tidia-ae, o professor pode determinar fluxo de atividades de forma dinâmica, o usuário professor pode realizar modificações, ou seja, adicionar e remover serviços, a qualquer momento do curso;
- iv) No Tidia-ae, o professor pode determinar fluxo de atividades. Cada usuário professor pode escolher a sequência das ferramentas que serão trabalhadas, atribuindo data de início e fim de uma atividade.

### 5.6.4. Conceito “percepção do ambiente” no Tidia-ae

No mapa geral, este conceito, são tratados aspectos de percepção de tarefas e atividades, individuais e do grupo, como também a percepção social dentro do ambiente. Sendo subdividido em: percepção social e percepção de tarefas.

Neste conceito, o Tidia-ae, para a percepção social, oferece o recurso “Participantes”, que fica localizado na página de cada curso. Assim, o usuário professor ou o usuário aluno pode visualizar os integrantes de cada curso. Na percepção de tarefas, o Tidia-ae possui o recurso “Announcements”, nele o usuário pode consultar todo tipo de avisos destinados a ele.

### 5.6.5. Conceito “customização do ambiente” no Tidia-ae

No mapa geral, este conceito abrange todas as mudanças que podem ocorrer na organização, conteúdo e interface do ambiente. Essa customização pode ser especificada como, interna com adaptação manual ou automática para usuário, de forma estática ou dinâmica e adaptação manual ou automática para instituições e departamentos; ou externa, com a integração manual ou automática de ferramentas e AVA. O Tidia-ae implementa um customização por usuário de forma manual e automática, pois tanto o ambiente se adapta por ator quanto o próprio usuário pode customizá-lo, criando seus próprios sites dentro do ambiente. Além disso, para o Tidia-ae permite integração com ferramentas externas, como por exemplo, o sistema acadêmico.

#### 5.6.6. Conceito “portabilidade para dispositivos” no Tidia-ae

No mapa geral, a portabilidade para dispositivos indica se o ambiente oferece suporte para ser acessado a partir de diferentes dispositivos. No entanto, no Tidia-ae não foi encontrado suporte para TV Digital.

#### 5.6.7. Conceito “internacionalização” no Tidia-ae

Este conceito, no mapa geral, refere-se à capacidade do ambiente de oferecer suporte para que seus cursos possam ser acessados em diferentes idiomas, possua formato para diferentes moedas e data/hora. O Tidia-ae oferece suporte para que seus cursos sejam acessados em diferentes idiomas.

#### 5.6.8. Conceito “serviços” no Tidia-ae

No mapa geral, o conceito serviços, trata dos recursos utilizados no desenvolvimento do curso, podendo ser classificados como, serviços internos ou externos. E esses ainda podem ser serviços administrativos, didáticos, de avaliação, de comunicação, individuais, de grupo, de inteligência.

Assim, o Tidia-ae possui serviços para todas as categorias dos serviços internos, menos os de inteligência. Para os serviços internos administrativos, os serviços que permitem que professor, possa gerenciar o seu curso, por exemplo, escaninho (usado na troca de documentos entre alunos e professores), quadro de notas, cronograma e conteúdo programático.

Para os serviços didáticos interativos, o Tidia-ae implementa o serviço “bate-papo”, “whiteboard” (permite que os usuário se comunique de forma gráfica através de um quadro) e

o “comunicador instantâneo”. Além de, também, implementar os não interativos, material didático on-line e webgrafia. Para avaliar os alunos os professores encontram no Tidia-ae, exercícios, tarefas, fóruns.

Nos serviços internos de comunicação o Tidia-ae implementa os serviços assíncronos, por exemplo, o serviço “Mensagens” que permite enviar e receber mensagens e “Discussão” um tipo fórum. Além de implementar uma diversidade de serviços síncronos, por exemplo, o serviço “bate-papo”, “whiteboard” e o “comunicador instantâneo”. Nos serviços individuais há o portfólio, que está dividido em particular: acessível somente a ele e público: compartilhado com todos os usuários do ambiente. Além do “profile” espaço no qual os usuários podem publicar informações pessoais, incluindo foto. No Tidia-ae o usuário possui o “meu site”, onde o usuário pode verificar seus compromissos, armazenar seus próprios documentos e criar outros sites.

#### 5.6.9. Conceito “grupos” no Tidia-ae

Este conceito, no mapa geral, trata da possibilidade de formação e organização de grupos entre os usuários do ambiente dentro do curso. Não foi encontrado suporte a esse conceito no Tidia-ae.

### 5.7. Análise comparativa entre ambientes

Os ambientes analisados neste trabalho, baseados no mapa criado, mostraram que sob um análise quantitativa, implementam boa parte dos conceitos. No entanto, sob uma visão qualitativa deixa várias lacunas, ou seja, quando analisado, nos ambientes, o desdobramento de cada conceito, poucos foram os conceitos implementados totalmente pelos ambientes. Para melhor visualizar está análise, foi criada uma tabela comparando alguns ambientes em relação a alguns conceitos e subconceitos.

A tabela 1 apresenta uma análise comparativa entre os ambientes avaliados anteriormente, com relação às principais características do mapa conceitual, também, demonstrados em algumas seções anteriores. A primeira coluna refere-se aos conceitos mais amplos no mapa e a segunda aos subconceitos. Sendo assim, como consta na tabela, criaram-se três categorias de análise: i) implementa totalmente as funcionalidades, ou seja, para aquele subconceito o ambiente implementa todas as funcionalidades requeridas por ele a primeira; ii) implementa parcialmente as funcionalidades, ou seja, o ambiente implementa o conceito, mas

apenas em partes, pois uma ou funcionalidade de algum subconceito não é atendida por ele;  
iii) não implementa nenhuma funcionalidade especificada no conceito.

No próximo capítulo será apresentada a arquitetura de software referência para ambientes virtuais de aprendizagem, desenvolvida com base no mapa conceitual discutido neste capítulo. Primeiro serão discutidos os requisitos não-funcionais, também utilizados no desenvolvimento da arquitetura,

Tabela 1 - Análise Comparativa entre os Ambientes

		Ambientes					
		Amadeus	Moodle	Caroline	Eureka	Tidia – ae	Sakai
Conceitos							
Atores	Atores Flexíveis		♣♣			♣♣	♣♣
Interface	Usabilidade	♣♣			♣♣	♣♣	♣♣
	Design	♣♣		♣	♣	♣♣	♣♣
Cursos	Flexível com escolha de ferramentas	♣	♣♣	♣	♣♣	♣♣	♣♣
	Flexível com fluxo de atividades	♣	♣		♣♣	♣♣	♣♣
	Percepção de tarefas	♣♣	♣♣	♣	♣	♣♣	♣♣
Percepção de ambiente	Percepção social	♣♣	♣♣	♣♣	♣	♣♣	♣♣
	Interna para Automática, dinâmica e por usuário	♣♣	♣♣	♣♣	♣♣	♣♣	♣♣
Customização do ambiente	Interna para Instituições/Departamentos		♣♣	♣	♣		♣
	Externa com integração automática de ferramentas	♣			♣		♣
	Externa com manual de ferramentas ou AVA	♣			♣		♣
Serviços	Comunicação	♣	♣♣	♣♣	♣	♣♣	♣♣
	Didáticos	♣♣	♣♣	♣♣	♣	♣♣	♣♣
	Administrativos	♣♣	♣♣	♣♣	♣♣	♣♣	♣♣
	Individuais	♣	♣	♣	♣	♣♣	♣♣
	de Inteligência						
Portabilidade de dispositivos	Portabilidade de dispositivos	♣					♣♣
Internacionalização	Data/Moeda		♣♣				♣♣
	Idioma		♣♣				♣♣
Grupos	Definição		♣				♣♣
	Interação						♣♣
	Controle de visibilidade						♣♣
	Serviços de Grupo		♣				♣♣
Legenda:							
	♣♣	Implementa totalmente o subconceito					
	♣	Implementa parcialmente o subconceito					
		Não implementa					

## **5.8. Síntese**

Para o trabalho apresentado nesta dissertação já existe um ambiente que apresenta uma característica diferente. Sendo este, uma referência para os trabalhos futuros.

O Redu é um ambiente educacional que segue a filosofia de redes sociais, tendo sido concebido para que estudantes e profissionais de ensino disponham de ambientes de armazenamento e resolução colaborativa de provas e visualização do desempenho. (MELO, 2010).

O REDU oferece suporte à colaboração, discussão e disseminação de conteúdo educacional. Assim são especificadas as características de um novo conceito de plataforma de ensino que estende a experiência do usuário em mídia social e com seus pares num contexto de rede social para aprendizagem (LIMA, 2011).

Como o mapa conceitual proposto nesta dissertação, por ser baseado em componentes, pode evoluir, o componente para redes sociais poderia ser incluído no mapa conceitual e, conseqüentemente, na arquitetura.

## **6. ARQUITETURA DE REFERÊNCIA PARA AMBIENTES VIRTUAIS DE APRENDIZAGEM**

Apresenta-se neste capítulo uma arquitetura de referência para ambientes virtuais de aprendizagem, cuja elaboração foi baseada no mapa conceitual e em requisitos não-funcionais.

Nas próximas seções, primeiramente, serão discutidos alguns requisitos não-funcionais utilizados no desenvolvimento da arquitetura proposta e necessários para a implementação de um AVA, a arquitetura de referência proposta e, finalmente o seu detalhamento utilizando o processo UML Components (Daniels e Daniels, 2000). Além da arquitetura em si, este capítulo também apresenta um sistema de recomendação de arquiteturas, implementado para guiar o desenvolvedor durante o processo de projeto arquitetural, auxiliando-o a personalizar tanto a arquitetura de referência projetada, quanto os componentes funcionais derivados do processo UML Components. Para ilustrar o funcionamento dessa solução, o Capítulo 6 apresentará um possível refinamento da arquitetura proposta, no contexto do AVA Sakai, utilizando para isso o sistema de recomendação desenvolvido como parte da solução arquitetural proposta. O Capítulo 6 também apresenta diretrizes de como utilizar o Saka como um arcabouço para realizar a arquitetura de software em código..

## 6.1. Requisitos de Qualidade da Solução Proposta

Do ponto de vista dos requisitos de qualidade do sistema, foi realizada uma análise dos requisitos não-funcionais definidos pela norma produzida pela *International Standards Organization ISO/IEC 9126* (ABNT, 2003), que refere-se ao a qualidade no desenvolvimento de software. A norma também trás métricas e um modelo de avaliação para a sua aplicação em um processo de avaliação de um produto de software. Em seguida, os requisitos foram classificados em três níveis de prioridade: prioridade alta, prioridade média e prioridade baixa. Vale ressaltar que tal classificação foi extraída das mesmas entrevistas utilizadas para o desenvolvimento do mapa conceitual.

Tais requisitos podem ser encarados como desafios a serem alcançados por sistemas educacionais, para atender de fato ao que se espera deles: apoio ao ensino de qualidade e em larga escala, isto é, que atenda de maneira facilitada as diferentes necessidades dos usuários de AVAs. Para ilustrar melhor os requisitos de qualidade, assim como a importância de cada um no contexto de educação, será apresentado um cenário exemplificando cada requisito.

### 6.1.1. Requisitos de Qualidade como Prioridade Alta

Como critério para a classificação, requisitos considerados essenciais, isto é, cuja implementação é considerada obrigatória para o domínio, foram classificados como prioridade alta. Os requisitos de qualidade com prioridade alta para o mapa de referência proposto foram:

- **Transparência de distribuição:** os aspectos da distribuição do ambiente devem ser ocultados dos usuários e das aplicações. Este requisito está ligado ao requisito de *Usabilidade* da ISO 9126, mais especificamente à sub-característica de *operacionalidade*. Um exemplo desse requisito no contexto de educação poderia ser o fato de um aluno ou professor poder acessar o AVA sem que seja exigido conhecimento específico sobre sua infraestrutura de rede e a localização física dos servidores.
- **Segurança de acesso:** o ambiente deve garantir autenticidade e segurança no acesso. De acordo com a ISO 9126, este requisito é visto como uma sub-característica ligada ao requisito de qualidade denominado *Funcionalidade*. Um exemplo prático da sua utilização seria o fato de apenas o professor ser capaz de adicionar conteúdo e/ou informar as notas dos alunos.

- **Escalabilidade:** é um requisito obrigatório em um ambiente virtual de aprendizagem, pois deve haver um grande número de usuários acessando simultaneamente. Este requisito está ligado ao requisito de *Eficiência* da ISO 9126, mais especificamente à sub-característica de *utilização de recursos*. Um cenário real para a educação será o caso do servidor de EaD da universidade estar no limite de sua capacidade de acessos simultâneos. E, por uma questão estratégica, a universidade necessitar aumentar ainda mais o número de vagas e cursos ofertados. Se o sistema for escalável, bastaria adquirir novos servidores para trabalhar em conjunto, de forma colaborativa, com o já existente. O software deve permitir escalar facilmente o número de usuários do sistema.
- **Flexibilidade:** o ambiente deve ser personalizável de acordo com o usuário. Este requisito está ligado ao requisito de *Portabilidade* da ISO 9126, mais especificamente à sub-característica de *adaptabilidade*. De forma indireta, este requisito também está ligado ao requisito de *Usabilidade* da ISO 9126, mais especificamente à sub-característica de *Atratividade*. Um exemplo dessa adaptação poderia ser o fato de, enquanto o AVA apresenta para um aluno a ferramenta de Fórum em destaque, para outro, pode destacar a ferramenta de bate papo (*chat*), de acordo com o perfil de uso de cada usuário.
- **Interoperabilidade:** o ambiente deve ser capaz de estabelecer comunicação com outros ambientes ou com outros aplicativos. De acordo com a ISO 9126, este requisito é considerado uma sub-característica do requisito de qualidade denominado *Funcionalidade*. Essa é, sem dúvida, uma das grandes tendências e um dos principais desafios para a área de AVAs: integração entre o AVA e o ambiente externo, tais como redes sociais e outros AVAs. Com um AVA interoperável, seria possível, por exemplo, visualizar diretamente no facebook lembretes e materiais disponibilizados pelo professor no próprio AVA.
- **Portabilidade:** o módulo interno do sistema deve ser acessível a partir dos sistemas operacionais Windows e Linux. Este requisito está ligado ao requisito de *Portabilidade* da ISO 9126, mais especificamente à sub-característica de *capacidade para ser instalado*. O cenário real, nesse caso, é bastante autoexplicativo: de acordo com as características da instituição de ensino e a disponibilidade dos seus recursos computacionais, os servidores podem ser instalados em dispositivos Windows, Linux ou híbridos, formados por servidores Windows e Linux.

#### 6.1.2. Requisitos de Qualidade como Prioridade Média

Requisitos considerados importantes, isto é, cuja implementação apesar de não ser obrigatório, o cliente normalmente está disposto a pagar mais por eles, foram classificados como prioridade média.

Os requisitos de qualidade com prioridade média para o mapa de referência proposto foram:

- **Confiabilidade (*reliability*):** o ambiente deve funcionar corretamente todo o tempo que estiver *on-line*. Este requisito está ligado ao requisito de *Confiabilidade* da ISO 9126, mais especificamente à sub-característica de *tolerância a falhas*. Um exemplo desse requisito no contexto educacional pode ser: caso o AVA informe que a tarefa foi submetida com sucesso, o aluno pode confiar que, de fato, ela está acessível pelo professor, com um alto grau de confiança. Caso ocorra algum erro no envio, esse erro deve ser detectado e informado corretamente ao usuário.
- **Disponibilidade:** é a fração do tempo em que o componente (ou sistema) está operacional. O ambiente deve estar sempre disponível para que o usuário possa acessá-lo. Este requisito também está ligado ao requisito de *Confiabilidade* da ISO 9126, mais especificamente à sub-característica de *recuperabilidade*. Um cenário real no contexto educacional pode ser o simples fato de faltar energia elétrica no servidor Web. Nesse caso, o sistema pode estar apto a selecionar outro servidor redundante em localização física diferente.
- **Usabilidade:** o ambiente deve ser agradável e fácil de usar, para os vários perfis de usuários esperados. Este requisito está ligado ao requisito de *Usabilidade* da ISO 9126, mais especificamente às sub-características de *atratividade* e *apreensibilidade*. Um exemplo real desse requisito poderia ser o fato do aluno poder escolher como será a sua interface gráfica: se baseada em menus ou se baseado na ideia de *workflow*, onde o aluno segue um roteiro pedagógico menos livre, onde a sequência de ferramentas a serem utilizadas foi previamente planejada e especificada.
- **Facilidade de evolução:** a lógica do negócio costuma ser alterada com frequência. O custo de se trocar algum componente específico do sistema deve ser minimizado. Além disso, por conta do requisito de disponibilidade, as atualizações devem ser aplicadas de maneira a minimizar o tempo que o sistema fica *off-line*. Este requisito está ligado ao requisito de *Manutenibilidade* da ISO 9126, mais especificamente à sub-característica de *modificabilidade*. Um cenário no contexto de educação poderia ser o caso onde a

funcionalidade de submissão de tarefas tem sua lógica alterada. Nesse exemplo fictício, o professor deseja que para que ao submeter a tarefa, o aluno receba um e-mail com link de confirmação. Apenas ao clicar nesse link, a tarefa é liberada e, de fato, submetida para apreciação do professor.

- **Refatoração:** O sistema deve facilitar a implantação de mudanças perfectivas no sistema, com o intuito de melhorar a sua qualidade estrutural e facilitar futuras manutenções evolutivas. Este requisito está ligado ao requisito de *Manutenibilidade* da ISO 9126, mais especificamente à sub-característica de *analísabilidade*. Um exemplo desse requisito de qualidade poderia ser a refatoração do módulo de comunicação com o Sistema de Gerenciamento de Banco de Dados (SGBD) MySQL, para que seja mais fácil, no futuro, substituí-lo por outro SGBD.
- **Reusabilidade:** O sistema deve facilitar o reuso sistemático de suas partes, com o intuito de facilitar e agilizar o desenvolvimento de outros sistemas semelhantes. Este requisito não é considerado diretamente pela ISO 9126, mas foi considerado importante, tendo em vista o impacto no custo e no tempo de desenvolvimento do produto. Um exemplo desse requisito de qualidade poderia ser a reutilização de ferramentas como Fórum, bate papo (*chat*) e portfólio entre dois AVAs distintos.

### 6.1.3. Requisitos de Qualidade como Prioridade Baixa

São requisitos que apesar de desejáveis, o cliente normalmente não está disposto a pagar mais por eles, foram classificados como prioridade baixa. Os requisitos de qualidade com prioridade baixa para o mapa de referência proposto foram:

- **Eficiência:** o tempo de resposta do ambiente para cada ação do usuário. Este requisito está ligado ao requisito de *Eficiência* da ISO 9126, mais especificamente à sub-característica de *comportamento em relação ao tempo*. Um exemplo direto na utilização desse requisito é o fato do aluno ter garantias de que sua mensagem será publicada no fórum em no máximo um minuto após a submissão.
- **Confinamento de falhas:** Caso ocorra alguma falha no ambiente, elas devem ser isoladas de modo que a correção deve ser feita apenas naquele módulo da falha. Este requisito está ligado a dois requisitos de qualidade da ISO 9126: (1) ao requisito de *Confiabilidade*, mais especificamente à sub-característica de *recuperabilidade*; e (2) ao requisito de *Portabilidade*, mais especificamente à sub-característica de *capacidade para substituir*.

## 6.2. Arquitetura de Referência da Solução Proposta

O projeto arquitetural da solução proposta objetivou inicialmente a especificação de uma arquitetura de referência que atendesse os requisitos de qualidade apresentados na seção anterior, respeitando em caso de conflito, a ordem de prioridade entre eles. Para isso, foi seguido um processo sistemático, baseado na ideia de catálogos de estilos e padrões arquiteturais, como apresentado na Figura 31. Para obter maiores informações sobre os catálogos de estilos e padrões arquiteturais consultar o Apêndice A.

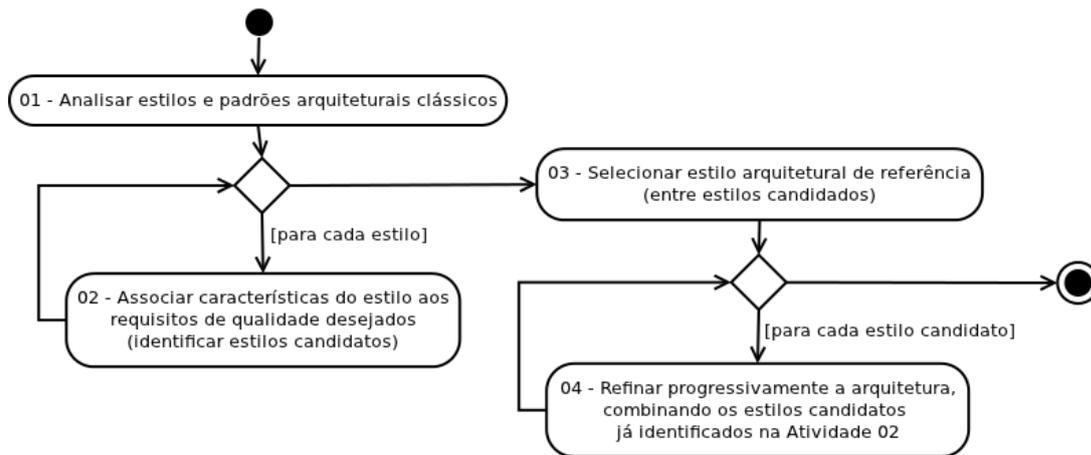
**Atividade 1:** Inicialmente, foram analisados estilos e padrões arquiteturais clássicos relatados na literatura (Shaw e Garlan, 1996), (Bass, Clements e Kazman, 2003), (Buschman *et al.*, 1996).

**Atividade 2:** Dadas as características de cada estilo e padrão arquitetural, foi feita uma associação entre as características de cada estilo e os requisitos de qualidade que os mesmos favorecem e prejudicam.

**Atividade 3:** Em seguida, foi selecionado um estilo arquitetural como referência básica para iniciar o projeto da arquitetura.

No contexto da solução proposta nesse trabalho, foi adotado o padrão arquitetural MVC como arquitetura inicial. A principal razão para adoção desse padrão como arquitetura inicial é o fato dele ser fortemente indicado para sistemas de informação (Buschman *et al.*, 1996), isto é, sistemas cujas funcionalidades normalmente envolvem consulta e atualização de banco de dados.

**Atividade 4:** Finalmente, foi realizado um refinamento progressivo dessa arquitetura inicial, com o intuito de combinar gradativamente outros estilos arquiteturais, de modo a melhorar o atendimento dos requisitos de qualidade.

**Figura 30 - Processo utilizado para projetar a arquitetura de referência**

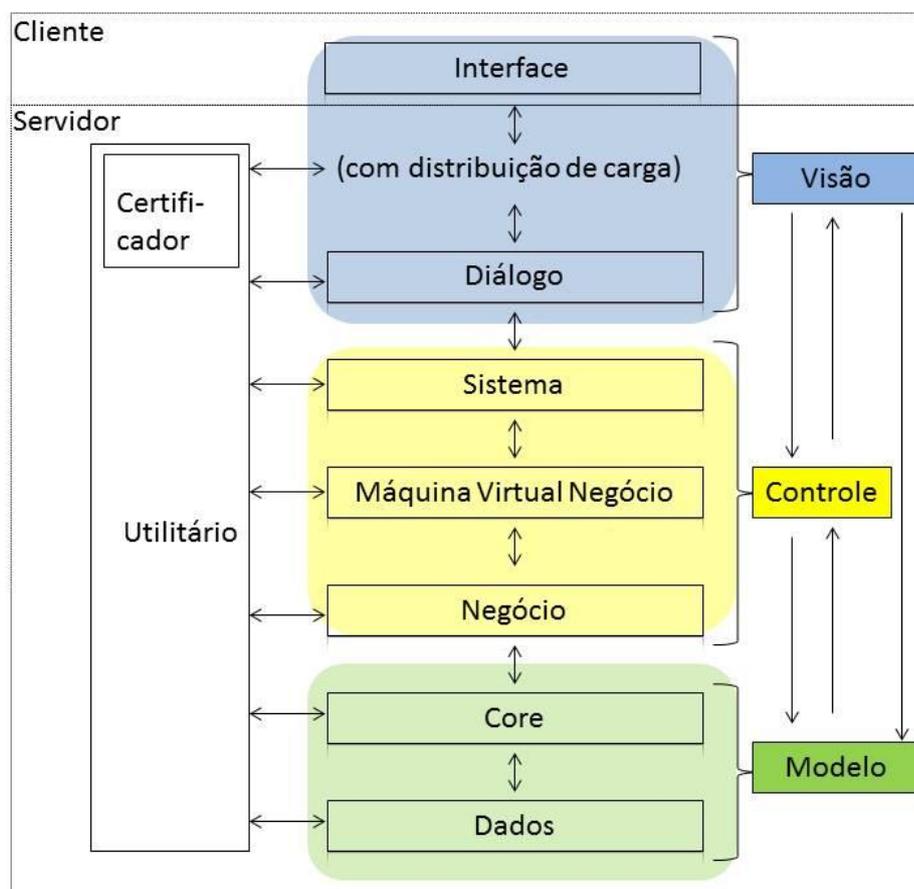
A Figura 32 apresenta a visão geral da arquitetura de referência final, fruto da execução do processo sistemático. Como pode ser percebido, a arquitetura definida possui um estilo arquitetural heterogêneo, baseado no MVC e que combina diversos estilos arquiteturais de maneira complementar. A seguir, é apresentada uma descrição de como, onde e porque cada estilo e padrão arquitetural foi utilizado na arquitetura de referência proposta:

**Padrão arquitetural MVC.** Esse padrão arquitetural foi adotado como base para iniciar o projeto da arquitetura de referência para AVAs. O padrão MVC apresenta três módulos básicos (Buschman et al., 1996):

- i) Model: responsável pelas entidades do domínio da aplicação (estrutura de dados);
- ii) View: responsável pela exibição dos dados; e
- iii) Controller: responsável pela execução da lógica do negócio”. Além do fato do padrão MVC ser recomendado para sistemas de informação (Buschman et al., 1996), a separação explícita entre o módulo de interface com o usuário (View) e o restante do sistema (Controller e Model) favorece a especificação de interfaces adaptativas, de acordo com o perfil do usuário.

Sendo assim, considerou-se que esse padrão arquitetural favorece a implantação do requisito de qualidade “Usabilidade”.

Figura 31 - Visão Geral da Arquitetura de Referência Proposta



- **Componentes Independentes.** Segundo Shaw e Garlan (1996) este estilo arquitetural representa o sistema através de um conjunto de processadores independentes (componentes), que se comunicam através de mensagens entre si.

Através da utilização deste padrão é possível obter uma alta escalabilidade, pois há “a possibilidade de se adicionar componentes gradativamente ao sistema, à medida que a demanda aumenta” (Brito, 2007, p. 17). Porém, para que esse potencial seja aproveitado, a arquitetura baseada em componentes independentes deve dispor de conectores apropriados que, quando necessário, desempenhem o papel de distribuidores de carga.

Outra potencialidade desse estilo arquitetural refere-se ao requisito de “Segurança de acesso”, mas para isso os conectores também possuem um papel de destaque, desempenhando papel de controle de acesso e criptografia centralizados. Finalmente, o fato dos componentes possuírem interfaces bem definidas e interagirem

entre si exclusivamente através dessas suas interfaces proporciona um baixo acoplamento. Esse baixo acoplamento beneficia direta e indiretamente outros requisitos de qualidade muito relacionados entre si: “Flexibilidade (funcionalidade flexível)”, “Facilidade de evolução” e “Confinamento de falhas”. Quanto menos acoplado for o sistema, mais fácil será trocar suas partes e, conseqüentemente, flexibilizar e evoluir suas funcionalidades.

Além do mais, por tratar-se de uma arquitetura de referência, outro fator que favorece a utilização desse estilo arquitetural é o fato dos componentes arquiteturais serem elementos abstratos que podem ser refinados durante o ajuste dessa arquitetura a diversas tecnologias de implementação, tais como classes e objetos, serviços e o próprio desenvolvimento baseado em componentes.

- **Camadas.** Apesar das arquiteturas de componentes independentes atuarem favoravelmente no tocante a “Facilidade de evolução”, ainda não fica claro quais os principais componentes que são mais voláteis e, portanto, potenciais alvos de manutenção e quais componentes são mais estáveis e, portanto, potenciais focos de reuso. Além do mais, a arquitetura precisa representar explicitamente como se dará interação com outras ferramentas externas ao AVA. Por essa razão, a arquitetura de referência proposta também foi influenciada pelo estilo arquitetural em camadas.

Como pode ser observado na Figura 31, basicamente, o estilo em camadas atuou em três aspectos complementares do sistema:

1) Definir uma camada específica para definir a forma de interação do sistema com o usuário que o está utilizando no momento (camada de *Diálogo*). Sendo assim, essa camada faz com que a arquitetura lide explicitamente com os requisitos de qualidade de “Usabilidade” e “Flexibilidade (usabilidade flexível)”;

2) Separar funcionalidades básicas do domínio (camada de *Negócio*) das funcionalidades específicas do sistema (camada de *Sistema*). Essa separação visa tanto promover reuso de software (“Reusabilidade”), quanto facilitar a evolução do sistema (“Facilidade de Evolução”). Isso é atingido graças ao fato dos componentes de *Negócio* serem mais estáveis e portanto, com maior chance de reuso e com menor incidência de evolução, ao contrário dos componentes e *Sistema*;

3) Além do mais, foi especificada uma camada voltada para sistemas legados (camada *Core*). Esta camada possibilita que funcionalidades existentes e implementadas em outros sistemas ou linguagens de programação sejam encapsuladas por funcionalidades básicas do *Negócio*. Essa representação explícita dos sistemas

legados facilita a integração com ferramentas externas, tais como redes sociais e outros AVAs e promove o requisito de qualidade de “Interoperabilidade”.

- **Máquina Virtual (ou Máquina Abstrata).** Apesar da arquitetura em camadas combinada com componentes independentes já atuar favoravelmente no tocante a “Facilidade de evolução”, para que essa evolução ocorra, ainda se faz necessário substituir componentes arquiteturais. Para facilitar a evolução de forma considerável e favorecer inclusive a atividade de implantação<sup>2</sup> dessas alterações. Como pode ser observado na Figura 31, a ideia geral por traz do estilo arquitetural de máquina virtual é:

1) localizar o ponto crítico da lógica de negócio; e

2) desacoplar essa lógica dos componentes do sistema, fazendo-a acessível por especialistas do domínio e não necessariamente de computação.

Nesse sentido, foi identificado que as alterações na lógica de negócio interferia diretamente na forma como os componentes da camada de *Sistema* requisitavam serviços dos componentes da camada de *Negócio*. Sendo assim, a conexão entre essas duas camadas foi identificada como ponto crítico. Nesse ponto crítico foi adicionada a camada de *Máquina Virtual*, que possui o papel de implementar as regras do negócio e armazená-las em uma base de regras. Sendo assim, as requisições de sistema são propagadas para a máquina virtual, que consulta a base de regras para saber quais serviços de negócio devem ser requisitados e em que ordem.

Uma mudança nas regras do negócio implicaria unicamente na atualização da respectiva regra, o que pode ser feito inclusive *on-line*, isto é sem que o sistema necessite ficar fora do ar por algum tempo. Dessa forma, o estilo arquitetural de Máquina virtual favorece tanto a “Facilidade de evolução”, quanto a “Disponibilidade” do sistema. Além do mais, outros requisitos de qualidade também são atendidos de forma indireta, tais como “Portabilidade” e “Flexibilidade”, uma vez que na hora de escolher os serviços de negócio, as regras podem considerar fatores como compatibilidade de plataforma e perfil do usuário.

No entanto, o nível de indireção provocado pela adição da máquina virtual e o tempo necessário para o processamento das regras pode prejudicar a o desempenho do sistema. Porém, como esse requisito foi considerado de prioridade baixa, optou-se por preservar a máquina virtual na arquitetura de referência. Além do mais, a utilização de

---

2 Do inglês *deployment*

um motor de inferência eficiente para o processamento das regras pode reduzir consideravelmente o tempo extra<sup>3</sup> requerido por esse estilo arquitetural.

Para ilustrar explicitamente como cada um dos requisitos não-funcionais são satisfeitos pela arquitetura especificada, a Tabela 2 apresenta uma síntese contendo, para cada um dos requisitos, os estilos arquiteturais que facilitam a sua satisfação e uma breve discussão de como são contemplados pela arquitetura.

---

3 Do inglês *overhead*

Tabela 2 - Mapeamento dos requisitos não-funcionais nos estilos

Requisito Não-funcional	Estilos Considerados	Discussão
Escalabilidade	Componentes independentes, cliente-servidor	Há possibilidade de se adicionar componentes gradativamente ao sistema, à medida que a demanda aumenta.
Transparência de distribuição	cliente-servidor	-
Segurança de acesso	Componentes independentes	Desempenha papel de controle de acesso e criptografia centralizados
Flexibilidade	Componentes independentes, Camadas	Quanto menos acoplado for o sistema, mais fácil será trocar suas partes e, conseqüentemente, flexibilizar e evoluir suas funcionalidades. A camada de <i>Diálogo</i> favorece a flexibilidade
Interoperabilidade	Camadas	-
Portabilidade	Máquina virtual	Quando os serviços de negócio são escolhidos, as regras podem considerar fatores como compatibilidade de plataforma e perfil do usuário.
Confiabilidade	cliente-servidor	-
Disponibilidade	Máquina virtual	Uma mudança nas regras do negócio implicaria unicamente na atualização da respectiva regra, o que pode ser feito inclusive <i>on-line</i>
Usabilidade	Padrão arquitetural MVC, Camadas	A camada de <i>Diálogo</i> favorece a Usabilidade
Facilidade de evolução	Componentes independentes, Camadas	Quanto menos acoplado for o sistema, mais fácil será trocar suas partes e, conseqüentemente, flexibilizar e evoluir suas funcionalidades.
Refatoração	Componentes independentes	-
Eficiência	-	-
Confinamento de falhas	Componentes independentes	Quanto menos acoplado for o sistema, mais fácil será trocar suas partes e, conseqüentemente, flexibilizar e evoluir suas funcionalidades

### 6.3. Detalhamento da Arquitetura Proposta em UML *Components*

A utilização do processo UML *Components* (CHEESMAN e DANIELS, 2000) adaptado, permite identificar os componentes internos das camadas da arquitetura. Sendo assim, nesta

seção será mostrado o detalhamento da arquitetura utilizando o diagrama de componentes UML.

Para a elaboração do diagrama de componentes com o detalhamento da arquitetura, o primeiro passo executado foi identificar as interfaces providas dos componentes. Para isso, foram processados os componentes do mapa conceitual e obedecida a arquitetura de referência. No entanto, nem todo conceito de mapa conceitual pode ser mapeado no diagrama de componentes UML, uma vez que os componentes arquiteturais, normalmente são responsáveis pelos requisitos funcionais do sistema (SHAW, 1996) e alguns conceitos do mapa conceitual referem-se a requisitos não-funcionais. Nesse caso, os conceitos não-funcionais já foram atendidos pela arquitetura de referência abstrata, como apresentado na Seção 5.2. Assim, o conceito serviços externos, se dará na arquitetura através de arquiteturas adaptadas. Para isso, poderão ser utilizados padrões de projeto, tais como Bridge ou Adapter (GAMA et al., 2000) para auxiliar na integração desses serviços aos demais componentes arquiteturais.

A customização do ambiente já está sendo atendida, em parte, pelo fato do detalhamento da arquitetura ser baseado em componentes e quanto à customização por usuário, só pode ser atendimento em nível de desenvolvimento de software. Esse atendimento é parcial porque para ser efetivo, se faz necessário preservar a rastreabilidade entre os elementos arquiteturais e o código fonte do sistema, durante a sua fase de implementação. O mesmo ocorre com o conceito “internacionalização” do mapa conceitual. Em relação aos subconceitos do conceito “interface”, eles precisam ser atendidos utilizando técnicas específicas que estão fora do escopo deste trabalho. No tocante ao requisito de portabilidade entre diferentes dispositivos, uma discussão mais detalhada é apresentada na Seção 6.1.1.

Vale salientar que a filosofia básica do processo UML Components é preservar o mapeamento entre requisitos e arquitetura de software. Nesse sentido, para cada um dos conceitos do mapa conceitual, deve haver uma interface provida contenha as operações relativas a ele. As Tabelas 3 e 4, apresentadas a seguir, apresentam as interfaces providas responsáveis pelas operações relativas aos conceitos do mapa conceitual de referência. Em seguida, a partir das interfaces identificadas, foram definidos componentes para oferecê-las de forma coesa, isto é, componentes que oferecem uma única interface, ou que agrupem apenas interfaces relacionadas. A Figura 33 apresenta a parte da arquitetura de software que foi refinada com os componentes arquiteturais identificados utilizando o processo UML *Components*. Buscando uma melhor legibilidade do diagrama de componentes as Figuras 34 e 35, apresentam o diagrama dividido em duas partes.

**Tabela 3 - Interfaces Providas pelos Componentes da Infraestrutura (Mapa Conceitual)**

<b>Interfaces da Infraestrutura (Mapa Conceitual)</b>	<b>Conceito do Mapa Conceitual que a Originou</b>
IGrupoMgr	Grupos
IComunicacaoMgr	Comunicação
ICursoMgr	Cursos
IDidaticoMgr	Serviços didáticos
ICadastrosMgr	Atores
IAdministrativoMgr	Serviços administrativos
IInteligenciaMgr	Serviços inteligência
IAvaliacaoMgr	Serviços avaliação
IIndividualMgr	Serviços individuais

**Tabela 4 - Tabela 3 - Interfaces Providas pelos Componentes da Aplicação (Mapa Conceitual)**

<b>Interfaces da Aplicação (Mapa Conceitual)</b>	<b>Conceito do Mapa Conceitual que a Originou</b>
IOpGrupo	Definição de Grupo Serviços de grupo Controle de Visibilidade
IOpComunicacao	Interação Inter-grupos Interação Intra-grupos Comunicação Percepção de Ambiente
IOpCurso	Padrão Flexível Fluxo de Atividades Escolha de Ferramentas
IOpDidatico	Serviços didáticos
IOpCadastros	Atores flexíveis Atores Padrão
IOpAdministrativo	Serviços administrativos
IOpInteligencia	Serviços inteligência

IOPAvaliacao	Serviços avaliação
IOPIndividual	Serviços individuais

Figura 33 - Diagrama de Componentes

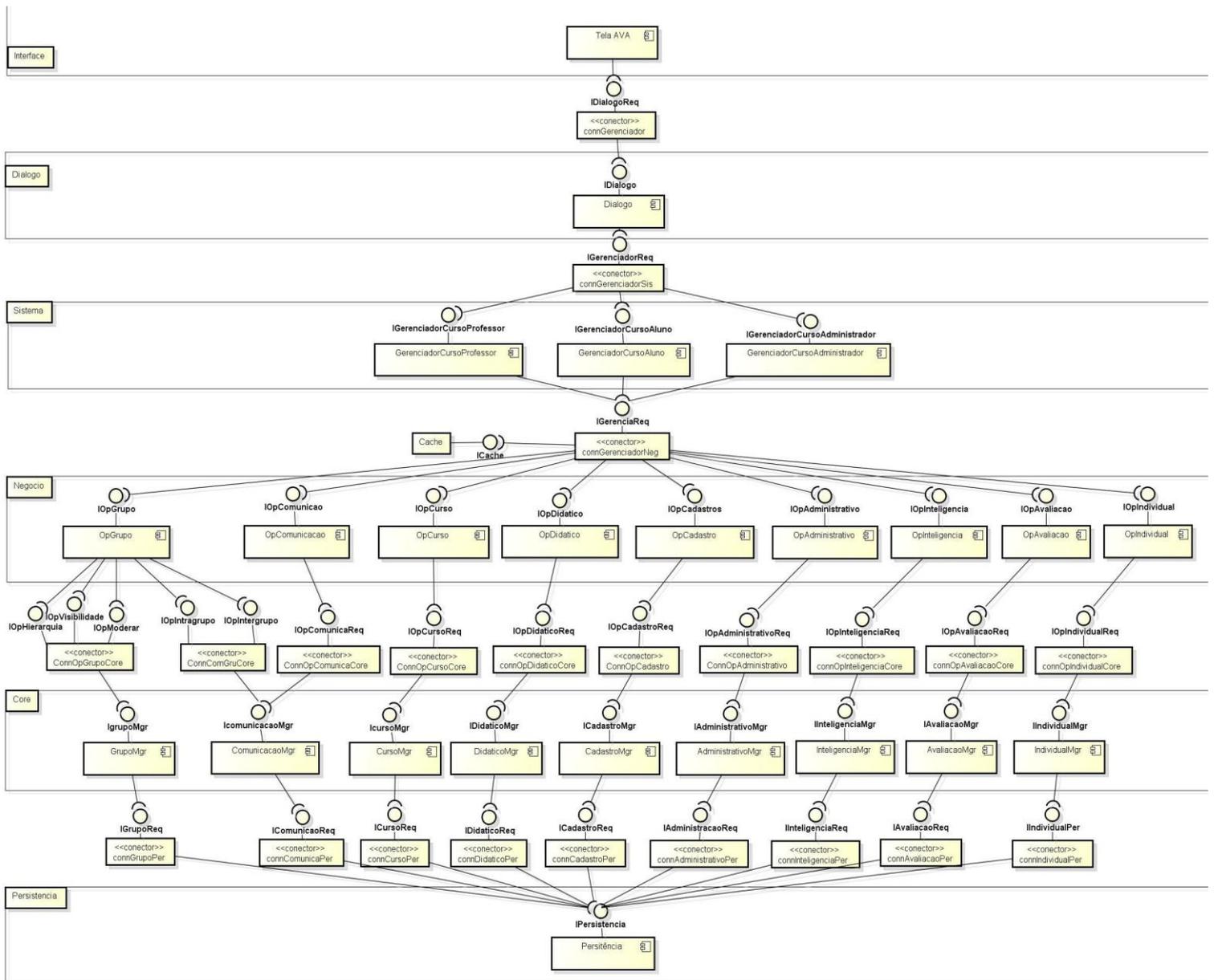
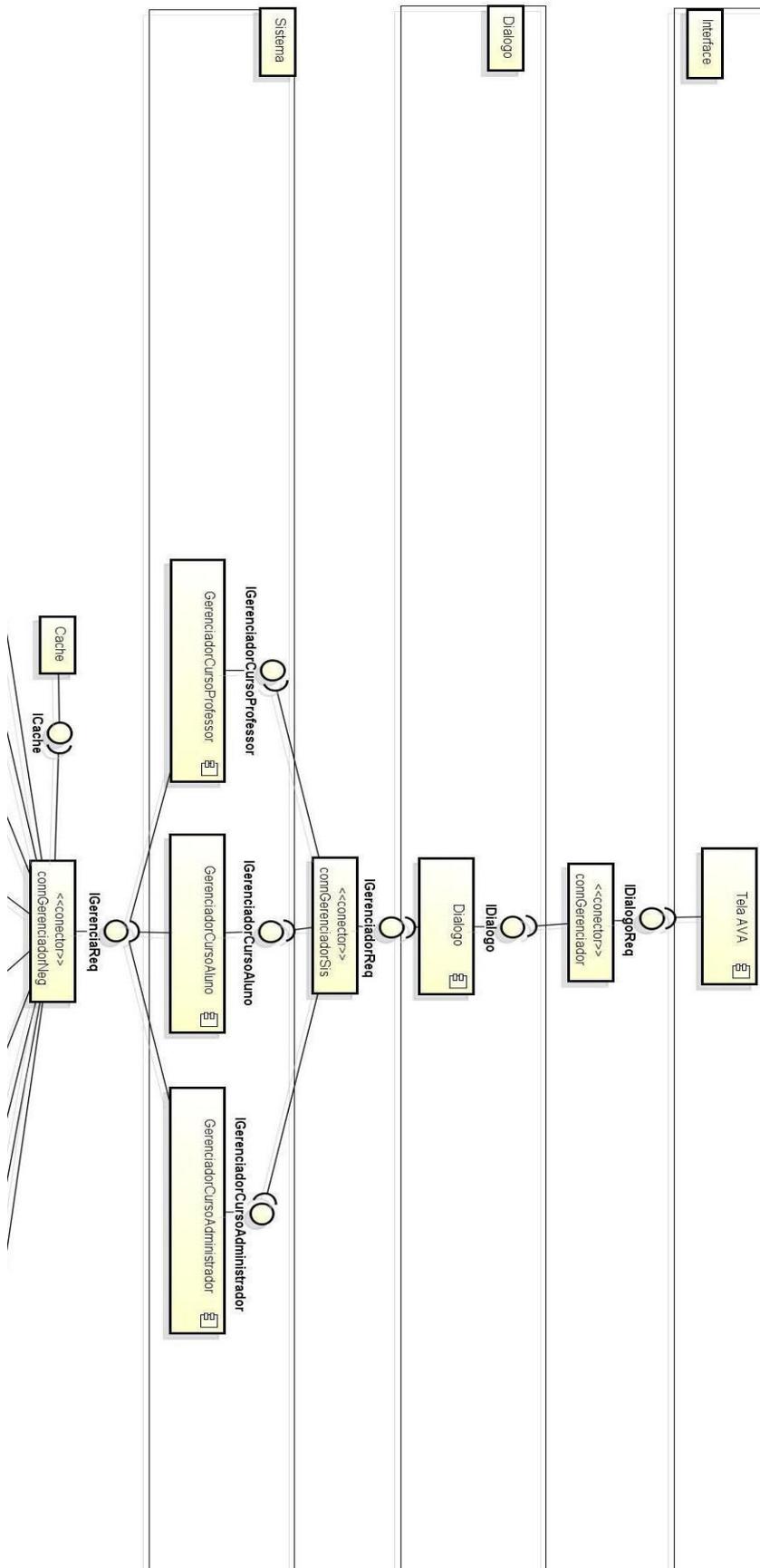
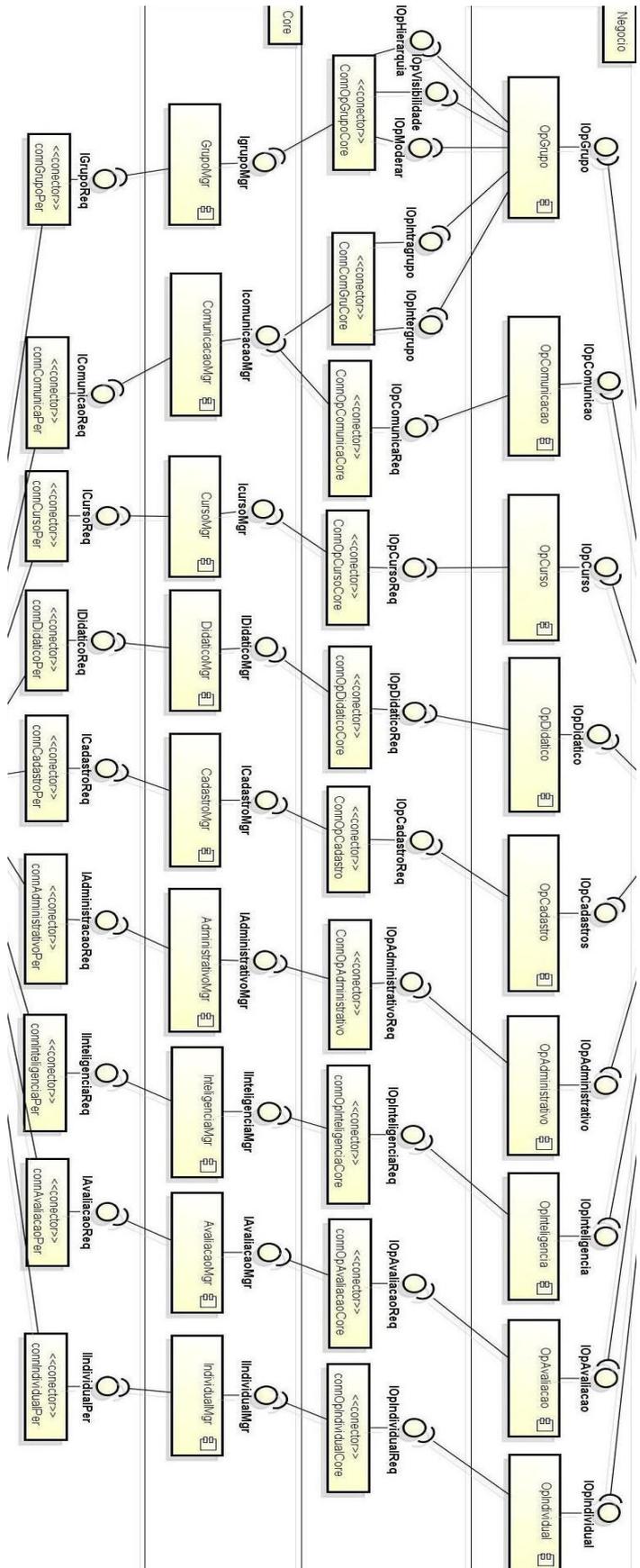


Figura 32 - Diagrama de Componentes - parte 1



**Figura 33 - Diagrama de Componente - parte 2**



#### 6.4. Sistema de Recomendação Proposto

Como já foi anunciado nas seções anteriores, o desenvolvimento do mapa conceitual justificou-se pela inserção do educador no desenvolvimento de um AVA que atendesse suas necessidades. Porém, apesar da relevância desse primeiro auxílio, ainda há uma distância considerável entre as decisões de preferência no mapa conceitual e as decisões de projeto na arquitetura de software. Por essa razão, nesta seção será apresentado um sistema de recomendação baseado em conhecimento, cujas regras foram extraídas do mapa conceitual apresentado no Capítulo 4 e dos requisitos não-funcionais apresentados na Seção 6.1. O objetivo desse sistema é facilitar o mapeamento entre as decisões do educador e o impacto disso para o desenvolvedor de software.

Ao ser executado, o sistema de recomendação apresenta uma série de perguntas, que deverão ser respondidas pelo usuário. As primeiras perguntas estão relacionadas aos requisitos não-funcionais, portanto a partir delas será gerada arquitetura similar a da Seção 6.2. No entanto a arquitetura resultante do sistema pode ter qualquer sub-conjunto de requisitos, dependendo do que foi respondido ao sistema. Após a definição da arquitetura abstrata, derivada da arquitetura de referência, as perguntas seguintes são voltadas aos conceitos relativos a requisitos funcionais e têm o propósito de gerar a arquitetura detalhada. Assim, baseado na base de conhecimento e nas informações passadas pelo usuário o sistema inferirá todos os componentes, conectores e interfaces que o diagrama de componentes deve possuir. Algumas regras da base de conhecimento do sistema podem ser observadas na Figura 43.

Então, uma vez recomendada à arquitetura e seus devidos detalhamentos, tal estrutura servirá como subsidio para o desenvolvedor durante as demais fases do desenvolvimento do AVA.

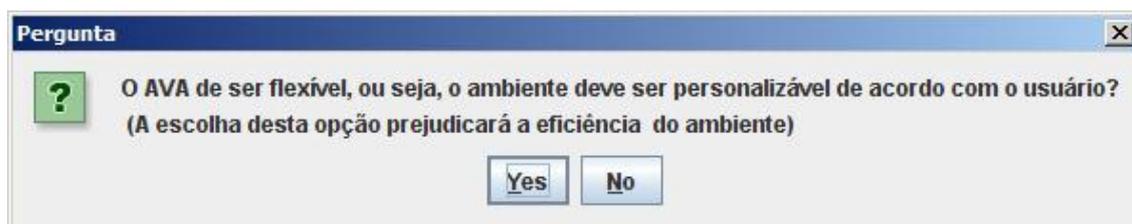
Para o desenvolvimento do sistema de recomendação foi utilizado o Inabit, um *framework* que oferece recursos para a criação de sistemas baseados em conhecimento. Sendo assim, a seguir, serão apresentadas algumas telas do sistema. Primeiramente, algumas telas que servem para recomendar a arquitetura e depois as que recomendam o detalhamento.

Figura 36 - Sistema de Recomendação Tela 1



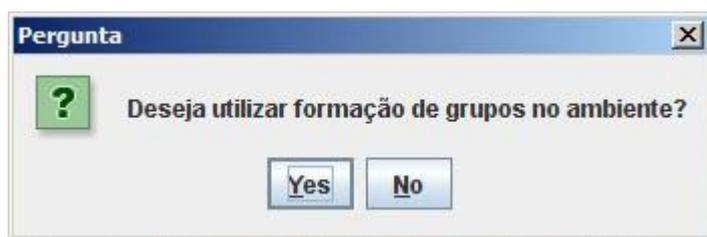
Na tela apresentada na Figura 36, o sistema deseja saber do usuário se deve acrescentar à arquitetura o requisito Facilidade de evolução da Seção 6.1.2 a arquitetura. Caso a resposta seja positiva, ele adiciona a camada de máquina virtual à arquitetura. Em outra pergunta, caso essa característica precise ser combinada com alto desempenho, à arquitetura também tem que conter um sistema de cache das regras de negócio contidas na máquina virtual. Esse cache é realizado pelo componente Cache junto com sua interface ICache.

Figura 34 - Sistema de Recomendação Tela 2



A tela apresentada na Figura 37 deseja saber do usuário se de deve adicionar o requisito Flexibilidade da Seção 6.1.2 a arquitetura.

Figura 35 - Sistema de Recomendação Tela 3



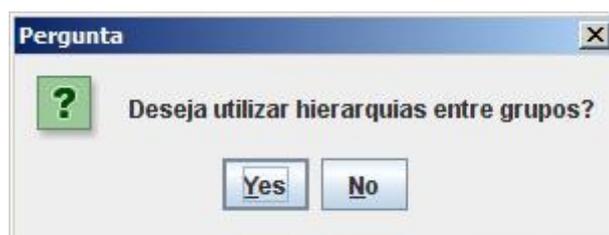
Para o diagrama de componentes, apresentado na Seção 6.3, existe uma estrutura base formada por componentes, conectores e interfaces. Essa estrutura contém alguns elementos que não são influenciados pelas escolhas do usuário em relação ao detalhamento da arquitetura. Sendo assim, tais elementos são sempre sugeridos para serem acrescentados a seguinte estrutura:

- **Componentes:** Persistencia, GerenciadorCursoAluno, GerenciadorCursoProfessor, GerenciadorCursoAdministrador, Dialogo e Tela AVA;

- **Conectores:** connGerenciadorNeg, connGerenciadorSis, connGerenciador;
- **Interfaces:** IDialogoReq, IDialogo, IGerenciadorReq, IGerenciadorCursoAluno, IGerenciadorCursoProfessor, IGerenciadorCursoAdministrador, IGerenciaReq, IPersistencia;

Quando a tela da Figura 38 é apresentada ao usuário o sistema que saber se deve acrescentar o recurso de grupo ao diagrama de componentes. Caso o usuário clique em “Yes”, o sistema adicionará os componentes GrupoMgr, OpGrupo, os conectores connGrupoPer e connOpGrupoCore e as interfaces IGrupoReq, IGrupoMgr, IOpGrupo à arquitetura detalhada. Caso o usuário responda “Yes” para a pergunta da Figura 39, será acrescentada a interface IOpHierarquia ao diagrama. Assim, será possível existir hierarquia entre grupos no AVA. No entanto, a pergunta da Figura 39 só será feita, se antes o usuário tiver respondido “Yes” para a Figura 38, uma vez que existe uma dependência entre esses conceitos do mapa conceitual.

**Figura 36 - Sistema de Recomendação Tela 4**



Outra consequente da tela da Figura 39 é a da Figura 40. Caso a resposta para a tela apresentada na Figura 40 seja “Yes”, será adicionada ao diagrama a interface “IOpVisibilidade”. Assim, para a resposta “Yes” na tela da Figura 41, também consequência da Figura 39, será adicionada a interface IOpModerar.

**Figura 37 - Sistema de Recomendação Tela 2**



Figura 38 - Sistema de Recomendação Tela 3



Figura 39 - Sistema de Recomendação Tela 4



A tela apresentada na Figura 42, mostra o resultado do sistema caso a resposta para as Figuras 38, 39 e 40 sejam positivas.

Figura 40 - Base de Conhecimento

```

ResultsWindow.java  MainTeste.java  recomendaAVA  AskToUser.java
OpComunicacao, ConnOpComunicaCore, ComunicacaoMgr, connComunicaPer,
IOpComunicacao, IOpComunicaReq, IcomunicacaoMgr, IComunicacaoReq,
GrupoMgr, OpGrupo, connGrupoPer, connOpGrupoCore, IGrupoReq, IGrupoMgr,
IOpGrupo, IOpHierarquia, IOpVisibilidade, IOpModerar : enumerated ("conector","interface","componente");

valid, true, false, comunicacao, hierarquia, visibilidade, moderar : enumerated("true", "false");

constants:
true = "true";
false = "false";
comunicacao = "true";
hierarquia = "true";
visibilidade = "true";
moderar = "true";

rules:

r0: if valid=true then true=ask("OpComunicacao","Deseja utilizar serviços de comunicação?", "componente");
r1: if OpComunicacao = "componente" then ConnOpComunicaCore="conector";
r2: if ConnOpComunicaCore = "conector" then ComunicacaoMgr ="componente";
r3: if ComunicacaoMgr ="componente" then connComunicaPer = "conector";
r4: if connComunicaPer = "conector" then IOpComunicacao = "interface";
r5: if IOpComunicacao = "interface" then IOpComunicaReq = "interface";
r6: if IOpComunicaReq = "interface" then IcomunicacaoMgr = "interface";
r7: if IcomunicacaoMgr = "interface" then IComunicacaoReq = "interface";
r8: if valid=comunicacao then comunicacao=ask("OpGrupo","Deseja utilizar formação de grupos no ambiente?", "componente");
r9: if OpGrupo = "componente" then GrupoMgr="componente";
r10: if GrupoMgr="componente" then connGrupoPer="conector";
r11: if connGrupoPer="conector" then connOpGrupoCore="conector";
r12: if connOpGrupoCore="conector" then IGrupoReq="interface";
r13: if IGrupoReq="interface" then IGrupoMgr="interface";
r14: if IGrupoMgr="interface" then IOpGrupo="interface";
r15: if IOpGrupo="interface" then hierarquia=ask("IOpHierarquia","Deseja utilizar hierarquias entre grupos?", "interface");
r16: if IOpGrupo="interface" then visibilidade=ask("IOpVisibilidade","Deseja que os grupos possuam controle de visibilidade?", "interface");
r17: if IOpGrupo="interface" then moderar=ask("IOpModerar","Deseja que apenas os professores possam criar grupos?", "interface");

```

A Figura 43, mostra algumas regras, criadas a partir do mapa conceitual, utilizadas nas inferências realizadas pelo *Inabit*.

## 7. REFATORAÇÃO DA ARQUITETURA DO AVA SAKAI UTILIZANDO A ABORDAGEM PROPOSTA

No Capítulo 4 foi apresentado o mapa conceitual de ambientes virtuais de aprendizagem, que serviu como requisitos funcionais para um AVA. No Capítulo 6 foram apresentados os requisitos-não funcionais, a arquitetura de referência, o detalhamento desta arquitetura em UML componentes e finalmente o sistema de recomendação baseado em conhecimento.

Neste capítulo será apresentado um cenário ilustrativo, utilizando o AVA *Sakai*, da interação do educador com o sistema de recomendação a fim de gerar uma arquitetura que atenda as necessidades de um determinado educador. Estando essa arquitetura acompanhada de um refinamento em UML *components*.

Mais especificamente, ilustrar o funcionamento da solução proposta num cenário real, é apresentado um possível *refinamento da arquitetura proposta, no contexto do AVA Sakai*, utilizando para isso o sistema de recomendação desenvolvido como parte da solução arquitetural proposta. Também são apresentadas diretrizes de como utilizar o Sakai como um arcabouço para facilitar a realização da arquitetura de software em código.

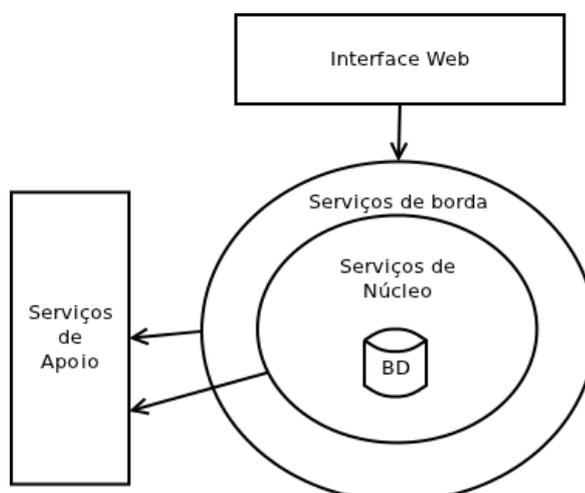
Por se tratar de uma arquitetura de referência, para que essa arquitetura seja utilizada no contexto de um AVA real, ela deve antes ser refinada sob a perspectiva de tecnologias de implementação e restrições estruturais, tais como limitações de arcabouço<sup>4</sup> e/ou bibliotecas a serem utilizadas. Nesta seção é apresentado o refinamento da arquitetura de referência apresentada na Seção 7.2, no contexto do AVA *Sakai* OAE.

Para compreender melhor o trabalho de refinamento a ser feito, a Figura 44 apresenta a arquitetura de software do AVA *Sakai*. Como pode ser visto, o *Sakai* adota um estilo arquitetural heterogêneo, que combina características de arquitetura orientada a serviços SOA “na qual funções chave (serviços) são construídas como componentes reutilizáveis que implementam padrões da indústria para comunicações interoperáveis”( Zanut, 2007, p.1) e arquitetura em camadas. Dessa forma, os serviços são estruturados em dois níveis: serviços de núcleo e serviços de borda. Os serviços do núcleo são considerados de sistema e devem ser utilizados por serviços da borda. Os serviços da borda, por sua vez, podem ser disponibilizados para os usuários na interface do AVA.

---

4 Do inglês *framework*

Figura 41 - Visão Geral da Arquitetura do Sakai



O primeiro passo para o refinamento da arquitetura de referência no contexto do *Sakai* é realizar a correspondência entre cada elemento da arquitetura do *Sakai* e da arquitetura de referência. Sendo assim, é possível fazer a analogia apresentada na Tabela 5. Analisando essa tabela, percebemos que o AVA *Sakai* não é totalmente compatível com a arquitetura de referência proposta. Sendo assim, para que os requisitos de qualidade desejados sejam preservados, precisaríamos implementar os elementos arquiteturais ausentes no *Sakai*. Porém, vale ressaltar que a resolução de inconsistências arquiteturais podem ser caras de se executar (Sommerville, 2006). Por exemplo, para a implementação do elemento arquitetural *Máquina Virtual Negócio*, será necessário interferir diretamente nos serviços de borda, com o intuito de extrair a lógica de negócio deles e evitar que eles evoquem diretamente os serviços de núcleo. Ao invés disso, deve ser feita uma requisição a uma nova categoria de serviços: os serviços de máquina virtual. A melhor forma de realizar a re-engenharia da arquitetura dos AVAs existentes para a arquitetura de referência proposta pode ser vista como mais um desafio.

Tabela 5 - Mapeamento entre os principais elementos da arquitetura de referência e do *Sakai*

Elemento Arquitetural de Referência	Elemento Arquitetural <i>Sakai</i>
Interface	Interface Web
Conector com distribuição de carga	-
Diálogo	-
Sistema	Serviços de borda
Máquina Virtual Negócio	-

Negócio	Serviços de núcleo
Core	-
Dados	Banco de Dados (BD)
Utilitário	Serviços de apoio

Nas próximas seções será apresentada a execução do sistema de recomendação para recomendar uma nova arquitetura para o AVA *Sakai* incluindo essa nova especificação. Além disso, será apresentada, também, a arquitetura recomendada com seus devidos o detalhamento no diagrama de componentes.

### 7.1. Execução do Sistema de Recomendação

Nesta seção é descrito um cenário para uma nova arquitetura do AVA *Sakai*. Como cenário imaginemos o seguinte: O educador ao submeter um questionário no AVA, deseja que o aluno receba um e-mail com link de confirmação. Apenas ao clicar nesse link, a tarefa é liberada e, de fato, submetida para apreciação do professor. No entanto, está funcionalidade de enviar um link por e-mail ainda não existe no AVA.

Com esse cenário é preciso alterar a regra de negócio do AVA. Surgindo com isso, a necessidade de um AVA fácil de evoluir (requisito discutido na Seção 6.1). Então, para atender a necessidade posta no cenário, o educador pode utilizar o sistema de recomendação. Assim, quando o sistema apresentar a tela da Figura 36, o usuário deve responder positivamente.

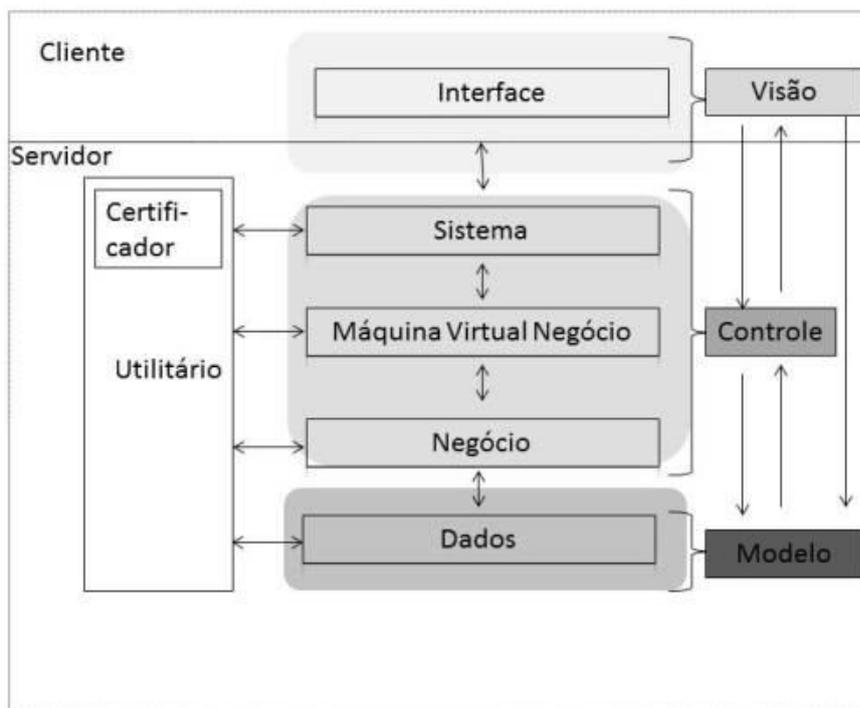
### 7.2. Arquitetura do AVA *Sakai* gerada pelo sistema de recomendação

Após realizar os demais questionamentos ao usuário, o sistema recomenda uma nova arquitetura para o *Sakai*. Assim, como discutido anteriormente, é adicionar a arquitetura do *Sakai* uma nova camada, a Máquina Virtual, pois esta possui o papel de implementar as regras do negócio e armazená-las em uma base de regras. Então, as requisições de sistema são propagadas para a máquina virtual, que consulta a base de regras para saber quais serviços de negócio devem ser requisitados e em que ordem. Então, a nova arquitetura recomendada é apresentada na Figura 45.

Caso ocorra uma mudança nas regras do negócio, a única implicação é atualizar a respectiva regra ou acrescentar uma nova, podendo ser feito on-line. Dessa forma, o estilo

arquitetural de Máquina virtual favorece a “Facilidade de evolução”, quanto a “Disponibilidade” do sistema.

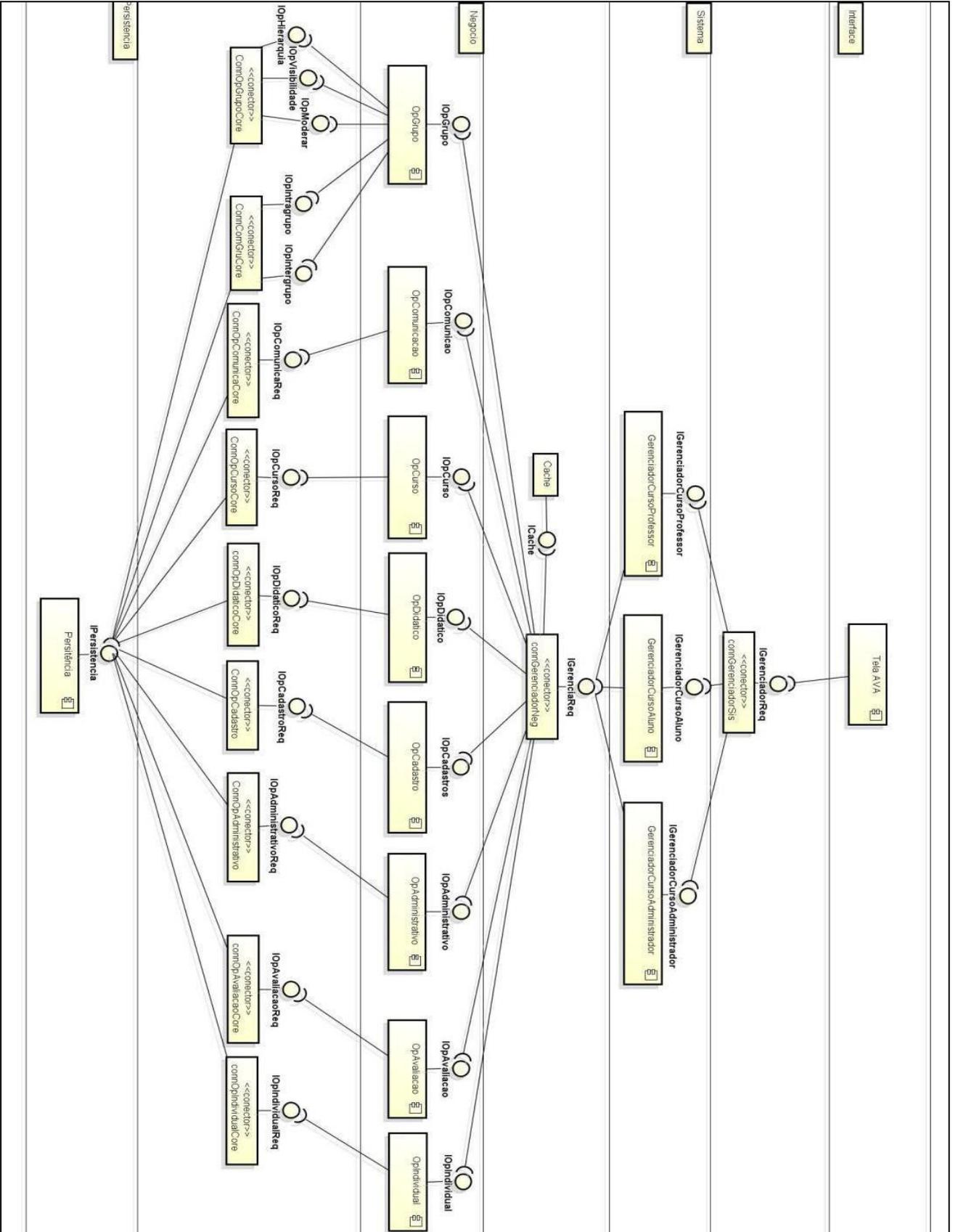
**Figura 42 - Arquitetura Recomendada para o Sakai**



### 7.2.1. Detalhamento da arquitetura

Para a arquitetura gerada na seção anteriormente foi gerada uma detalhada, que pode ser visualizada na Figura 46.

Figura 43 - Diagrama de Componentes Sakai



## 8. CONCLUSÃO E PERSPECTIVAS FUTURAS

Um dos mais importantes objetivos deste trabalho foi aproximar mais o educador do desenvolvimento do AVA que ele irá utilizar. Assim, para que isso fosse possível, neste trabalho foi apresentado, no Capítulo 4, um mapa conceitual contemplando características dos principais ambientes virtuais de aprendizagem. Assim, este mapa conceitual pode servir de guia para obtenção de requisitos funcionais para uma grande combinação de potenciais AVAs. No Capítulo 6 foram apresentados os principais requisitos-não funcionais para a construção de AVAs, que foram obedecidos na elaboração da arquitetura de referência. O detalhamento desta arquitetura em diagrama de componentes foi baseado nos requisitos funcionais definidos no mapa conceitual.

Em relação ao mapa conceitual concebido, buscou-se dar uma visão em amplitude dos conceitos de um AVA, pois a verticalização, ou seja, os detalhamento em subconceitos não são suficientes para descrever cada conceito. Assim, por exemplo, o conceito interface possui detalhamentos que poderiam dar origem a outro trabalho mais específico. Além disso, não se espera que este trabalho consiga abranger todas as possíveis características de um AVA, uma vez que o universo destes ambientes é muito extenso, sendo este um dos motivos para que o mapa conceitual fosse elaborado com a composição de conceitos. Com isso, caso surja alguma nova característica ou deseje-se considerar outros AVAs, o modelo conceitual pode ser evoluído, assim como a solução arquitetural proposta. Vale ressaltar que o impacto arquitetural após uma adição, remoção ou alteração de um conceito é considerado baixo, uma vez que só afetaria a fase de detalhamento dos componentes apresentada na Seção 6.3.

O mapa conceitual foi desenvolvido com o objetivo de se conceber uma visão do AVA para o educador e a arquitetura de referência seria a visão do desenvolvedor. Visando aproximar a visão do educador com a visão do desenvolvedor, foi desenvolvido um sistema de recomendação que mapeia as escolhas do educador, em relação às funcionalidades do AVA, em o que o desenvolvedor precisa para desenvolvê-lo.

Quanto à arquitetura de referência proposta, ela foi desenvolvida de uma forma que suporta qualquer combinação de requisitos não-funcionais. Por exemplo, caso o requisito não-funcional escalabilidade seja de baixa prioridade ou não seja necessário, pode-se remover da arquitetura a distribuição de carga. Bastando para uso, substituir um elemento arquitetural de forma localizada.

A camada de máquina virtual é outra característica importante da arquitetura. Utilizando máquina virtual é possível facilitar a escolha de ferramentas, além de flexibilizar o código do negócio. Conseqüentemente, viabiliza uma maior personalização de AVAs.

Um possível trabalho futuro para a solução proposta nesta dissertação é criar uma linha de produto de software. Assim, a partir das escolhas feitas pelo educador no sistema baseado em conhecimento o AVA seria criado automaticamente. Outro trabalho futuro seria a utilização de lógica Fuzzy nas inferências do sistema de recomendação, para que com isso fosse possível inferir um ordem de prioridade nos requisitos não- funcionais.

## Referências

- ALMEIDA, Maria Elizabeth Bianconcini. **Educação a distância na internet: abordagens e contribuições dos ambientes digitais de aprendizagem**. 2003. Disponível em: < <http://www.scielo.br/pdf/ep/v29n2/a10v29n2.pdf> >. Acesso em: 08 agosto 2011.
- AMADEUS. Disponível em:< <http://amadeus.cin.ufpe.br/index.html>>. Acessado em: 21 fev. 2011.
- BASS, Len; CLEMENTS, Paul; and KAZMAN, Rick. **Software Architecture in Practice (SEI Series in Software Engineering)**, 2nd Edition. Addison-Wesley, 2003.
- BRACHT, V. **A constituição das teorias pedagógicas da educação física**. Cadernos CEDES, Campinas, ano XIX, v. 19. n. 48, ago., 1999, p. 69-88.
- BRITO, Patrick Henrique da Silva et al. **Um processo para o desenvolvimento baseado em componentes com reuso de componentes**. Relatório Técnico, Instituto de Computação, 2007.
- BURKE, R. Hybrid. **Recommender Systems. The Adaptive Web: Methods and Strategies of Web Personalization**, Berlin-Heidelberg, v.4321, p.377-408, 2007.
- BUSCHMANN, Frank; MEUNIER, Regine; ROHNERT, Hans; SOMMERLAD, Peter; and Stal, Michael. **Pattern-Oriented Software Architecture**, Volume 1: A System of Patterns. Wiley, 1996.
- CAMPOS, K. **O Ambiente Virtual Eureka: Um estudo de caso da utilização em turmas de dependência do sistemas Matice pelos professores de graduação da PUCPR**. Curitiba . 2008. 155 f. Dissertação (Mestrado em Educação) – Centro de Tecnologia e Ciências Humanas. Universidade Pontifícia Católica do Paraná.
- CAZELLA, Sílvio César; NUNES, Maria Augusta S. N.; REATEGUI, Eliseo Berni. **A Ciência da Opinião: Estado da arte em Sistemas de Recomendação**, 200-?. Disponível em: <<http://www.dcomp.ufs.br/~gutanunes/hp/publications/JAI4.pdf>>. Acesso em: 16 julho 2011.

CHEESMAN, John and DANIELS, John. **UML Components: A Simple Process for Specifying Component-Based Software**. Addison-Wesley, 2000.

CHEN, Feng; WANG, Qianxiang; MEI, Hong; YANG, Fuqing. **An architecture-based approach for component-oriented development**. 26th Annual International Computer Software and Applications Conference, August 2002.

CLAROLINE, Disponível em: < <http://www.claroline.net/breve-presentation/>>. Acesso em: 20 maio 2011.

CRESPO, Sergio; FONTOURA, Marcus Felipe M. C. da; LUCENA, Carlos José P. de. **Um Modelo Conceitual Compatível com a Plataforma EDUCOM/IMS para Comparação de Ambientes de Educação na WEB**. 1998. Disponível em: < <http://fontoura.org/papers/sbie98.pdf> >. Acesso em: 24 abril 2011.

DAOLIO, J. **Educação física escolar: em busca da pluralidade**. Rev. Paul. Educ. Fís., São Paulo, supl. 2, 1996, p. 40-42.

EUREKA. Disponível em <http://eureka.pucpr.br/entrada/index.php>. Acessado em 07 abr. 2011.

FERBER, Stefan; HAAG, Jrgen; SAVOLAINEN, Juha. **Feature interaction and dependencies: Modeling features for reengineering a legacy product line**. In Proc. of the Second International Software Product Lines Conference (SPLC), LNCS 2379, pages 37–60, 2002.

FERREIRA, A. **Análise da Usabilidade no Ambiente Virtual de Aprendizagem Moodle**. BELÉM. 2007. 51 f. Monografia (Graduação em Engenharia da Computação) – Instituto de Estudos Superiores da Amazônia.

FILHO, Ivanildo José de Melo; GOMES, Alex Sandro; CARVALHO, Rosângela Saraiva e MELO, Rosangela Maria de. **Percepção social em EAD: Identificando necessidades para o LMS Amadeus**. Revista Brasileira de Informática na Educação, volume 1, número 1, dez. 2011. Disponível em: <<http://www.br-ie.org/pub/index.php/rbie>> Acesso em: 11abr. 2012.

FRANCISCATO, Fábio Teixeira et al. **Avaliação dos Ambientes Virtuais de Aprendizagem Moodle, TelEduc e Tidia - Ae: um estudo comparativo**. Revista Novas Tecnologias na Educação Vol. 6, p. 1-10, 2008. Disponível em: < <http://seer.ufrgs.br/renote/article/viewFile/14509/8428> >. Acesso em: 07 julho 2011.

GABARDO, Patricia; QUEVEDO, Silvia R. P. de e ULBRICHT, Vânia Ribas. **Estudo Comparativo das Plataformas de Ensino-Aprendizagem**. Encontros Bibli: revista eletrônica de biblioteconomia e ciência da informação, Florianópolis, volume 16, número 32. 2010. Disponível em: < <http://www.periodicos.ufsc.br/index.php/eb/index> > Acesso em: 11 abr. 2012.

GAMMA, Erich et al. **Padrões de Projeto: soluções reutilizáveis de software orientado a objetos**. Porto Alegre: Bookman, 2000

HAGUENAUER, C.J. e Victorino, A.L.Q. (2008) “**Avaliação em Educação a Distância Apoiada por Ambientes Virtuais de Aprendizagem**”. In: Revista Educaonline, v. 2.

KENSKI, Vani Moreira. **Das salas de aula aos ambientes virtuais de aprendizagem**. 2005. Disponível em: < <http://www.abed.org.br/congresso2005/por/pdf/030tcc5.pdf> >. Acesso em: 08 agosto 2011.

LIMA, I. **Análise das Práticas Docentes de Planejamento e Mediação em Redes Sociais no Ensino Médio**. Recife. 2011. 142 f. Dissertação (Mestrado em Ciência da Computação) – Centro de Informática. Universidade Federal de Pernambuco.

MACHADO, Suelen Fernanda, TERUYA, Teresa Kazuko. **MEDIAÇÃO PEDAGÓGICA EM AMBIENTES VIRTUAIS DE APRENDIZAGEM: A PERSPECTIVA DOS ALUNOS**. IX Congresso Nacional de Educação. 2009.

MELO, Cássio de Albuquerque. **Scaffolding of Self-Regulated Learning in Social Networks**. Recife. Recife. 2010. Dissertação (Mestrado em Ciência da Computação) – Centro de Informática. Universidade Federal de Pernambuco.

MONROE, Robert T.; KOMPANEK, Andrew; MELTON, Ralph; GARLAN, David. **Architectural styles, design patterns, and objects**. IEEE Softw., 14(1):43–52, 1997.

MOODLE. Disponível em <http://moodle.org/login/index.php>. Acessado em 14 abr. 2011.

MOREIRA, Jonathan Rosa. **Usabilidade, Acessibilidade e Educação a Distância**. 2011. Disponível em: < <http://www.abed.org.br/congresso2011/cd/13.pdf> >. Acesso em: 07 fevereiro 2012.

MOZZAQUATRO, Patricia; MEDINA, Roseclea. 2008. **Avaliação do Ambiente Virtual de Aprendizagem Moodle sob diferentes visões: aspectos a considerar**. Disponível em: < <http://seer.ufrgs.br/renote/article/viewFile/14508/8427>>. Acesso em: 19 agosto 2011.

OLIVEIRA, Anelise de Moraes et al. **Análise do ambiente virtual MOODLE como tecnologia de apoio aos estudantes de biblioteconomia**. *Múltiplos Olhares em Ciência da Informação*, volume 1, número 1.2011. Disponível em: < <http://portaldeperiodicos.eci.ufmg.br/index.php/moci/issue/current>> Acesso em: 11 abr. 2012.

OLIVEIRA, Leonardo Gomes de. **Proposta de uma estrutura metodológica para implementação de sistemas de recomendação**. 200-?. Disponível em:< [http://www.intellog.net/ArtigosNoticias/Arquivos/artigo\\_leonardo\\_oliveira.pdf](http://www.intellog.net/ArtigosNoticias/Arquivos/artigo_leonardo_oliveira.pdf)>. Acesso em: 07 julho 2011.

OLIVEIRA, LÍlian Simão; PIMENTEL, Maria da Graça; QUEIROZ-NETO, José Pinheiro. **The Digital TV as another solution to educate in isolated areas in the Amazon State, Brazil**. 2009. Disponível em: < [http://www.ufam-automation.net/idtvec/acceptedpapers/W1\\_11\\_oliveira.pdf](http://www.ufam-automation.net/idtvec/acceptedpapers/W1_11_oliveira.pdf)>. Acesso em: 08 julho 2011.

Resnick, P. & Varian, H. (1997) "Recommender Systems" *Communications of the ACM* v.40 p.56-58.

REZENDE S. *Sistemas Inteligentes Fundamentos e Aplicações*. Editora Manóle, 2003.

RIBEIRO, Elvia Nunes; MENDONÇA, Gilda Aquino de Araújo e MENDONÇA, Alzino Furtado. (2007). **A importância dos Ambientes Virtuais de Aprendizagem na busca de novos domínios na EAD**. Disponível em: <<http://www.abed.org.br/congresso2007/tc/4162007104526AM.pdf>>. Acesso em: 11abr. 2012.

SAKAI, **Collaboration and Learning Enviroment for Education**, Disponível em: < <http://sakaiproject.org/>>. Acesso em: 24 abril 2011.

SARMENTO, Wellington W. F. et al. Avaliação de Usabilidade no processo de desenvolvimento contínuo em Ambientes Virtuais de Aprendizagem: um estudo de caso com o ambiente SOLAR . 2011. In: Artigo publicado nos anais do XVI Simpósio Brasileiro de Informática na Educação.

SAVIANI, D. **Escola e democracia: teorias da educação, curvatura da vara, onze teses sobre a educação política**. 34.ed. Campinas, SP: Autores Associados, 2001.

SCHELEMMER, E. (2005) Cap. 9 - **Ambiente Virtual de Aprendizagem (AVA): uma proposta para a sociedade em rede de cultura de aprendizagem**, In VALENTINI, Carla Beatris e SOARES, Eliana M. S. (2005) Aprendizagem em ambientes virtuais: compartilhando ideias e construindo cenários p.135-159

SHAW, Mary; GARLAN, David. **Software Architecture: Perspectives on an Emerging Discipline**. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1996.

SILVA, J. **Uso de Mecanismos de Percepção da Tarefa para auxiliar os alunos no acompanhamento das atividades em Ambientes LMS**. Recife. 2010. 102 f. Dissertação (Mestrado em Ciência da Computação) – Centro de Informática. Universidade Federal de Pernambuco.

SILVA, Luiz Augusto Matos da; BARRETO, Luciano Porto. **Interoperabilidade de Unidades de Aprendizagem do IMS Learning Design em Ambientes Virtuais de Aprendizagem**. 2008. In: Artigo publicado nos anais do XIX Simpósio Brasileiro de Informática na Educação.

SOMMERVILLE, Ian. **Software Engineering**, 8th Edition. Addison-Wesley, 2006.

SWEENEY, Alan. **Agile e Open E-Learning Systems Architecture**. Athabasca. 2008. Dissertação (Master of Science in Information Systems). Athabasca University.

TIDIA AE, Disponível em: < <http://agora.tidia-ae.usp.br/portal>>. Acesso em: 20 maio 2011.

TORRES, Patrícia Lupion et al. SAAW: **A experiência da PUCPR no desenvolvimento de objetos de aprendizagem**. *Colabor@ - Revista Digital da CVA - Ricesu*, volume 5, número 19.2009.

ZANUZ, Luciano; FILIPPETTO , Alessandro S.; CRESPO, Sergio. 2007. Disponível em: <<http://www.seminfo.com.br/anais/2007/pdfs/6-35351.pdf>>. Acesso em: 07 fevereiro 2012.

GIL, Antônio Carlos. Como elaborar projetos de pesquisa. São Paulo. Atlas. 1991.

ABNT, NBR ISO/IEC 9126-1 **Engenharia de software - Qualidade de produto Parte 1: Modelo de Qualidade**, ABNT – Associação Brasileira de Normas Técnicas (2003).

QUIVY, Raymond ; VAN CAMPENHOUDT, Luc — **Manual de Investigação em Ciências Sociais**. Lisboa: Gradiva, 2008

## **APÉNDICE**

## APÊNDICE A – Padrões Arquiteturais de softwares

O papel de um padrão arquitetural é fazer uma associação entre os estilos arquiteturais (e combinações de estilos) e as suas respectivas aplicabilidades. A seguir, é apresentada uma relação dos principais estilos arquiteturais encontrados atualmente.

### 1. Baseada em fluxo de dados

- Descrição: Estilo onde o processamento é feito sequencialmente, isto é, um componente do sistema só executa após o seu antecessor finalizar a execução por completo.
- Problema: Sistemas onde o processamento deve ser realizado em partes sequenciais.
- Vantagens: (i) o fluxo de comunicação entre os componentes é relativamente simples e facilmente analisável; (ii) expressa uma arquitetura reconfigurável; (iii) expressa explicitamente a ideia de concorrência.
- Desvantagens: (i) unidirecional; (ii) indicado para sistemas interativos; (iii) exige muitos estágios de conversão e serialização de dados; (iv) o compartilhamento de dados globais é muito caro.
- Variantes:
  - *Pipes and filters*; Nessa variação, cada componente do sistema desempenha uma parte da transformação e em seguida, a sua saída é utilizada como entrada da próxima etapa do processamento (em outro componente)
  - Execução *batch*.

### 2. Centrado em Dados

- Descrição: Em uma arquitetura centrada em dados, o comportamento do sistema é especificado em função dos dados que circulam nele. Em outras palavras, nesses sistemas a lógica do negócio é definida em função da variação do estado do sistema.
- Problema: Sistemas onde há a necessidade de existir dados compartilhados.
- Vantagens: (i) facilidade de cooperação entre os componentes (maior integridade); (ii) separação entre os dados e o processamento (maior *modificabilidade*).

- Desvantagens: (i) não é possível encapsular os dados e o processamento juntos, os dois devem ser tratados obrigatoriamente (menor segurança); (ii) compreensão e *manutenibilidade* baixas, devido à dependência entre a lógica do negócio (processamento) e os dados.

- Variantes:

- Repositório – MVC

- *Blackboard*;

- *Publish/Subscribe*.

### 3. MVC

- Descrição: É um caso especial do repositório, que é uma variação do estilo arquitetural centrado em dados. No MVC existem três módulos principais no sistema: (i) *Model*: responsável pelas entidades do domínio da aplicação (estrutura de dados); (ii) *View*: responsável pela exibição dos dados; e (iii) *Controller*: responsável pela execução dos eventos (implementação das funcionalidades) e por avisar o *view* sobre modificações no *model*. Pode ser considerado uma particularidade especial de estilo em camadas. Porém, nesse caso, o *controller* deve desempenhar o papel de intermediário entre o *view* e o *model*.

- Problema: Transparência de localização entre os dados, o processamento da lógica do negócio e a exibição dos resultados.

- Vantagens: (i) separação entre os dados (*model*), a visualização (*view*) e o processamento (*controller*), o que acarreta numa melhor *manutenibilidade*.

- Desvantagens: (i) o desempenho pode ser comprometido, devido ao nível de indireção existente; (ii) nem todos os sistemas são mapeados facilmente com MVC.

### 4. Call/return

- Descrição: A comunicação de uma arquitetura *call/return* acontece a partir da chamada e retorno de funções. Dessa forma, cada componente do sistema é considerado um módulo funcional, que solicita serviços de outros módulos e responde aos serviços solicitados a ele. Devido à similaridade com o funcionamento das rotinas de programação atuais, o *call/return* é o estilo arquitetural mais adotado atualmente.

- Problema: Sistemas que não são centrados em dados, que necessite ser decomposto/*modularizado* e que tenha a figura do controlador de execução, isto é, um módulo responsável por orquestrar a execução e tratar os retornos dos módulos envolvidos.
- Vantagens: (i) uni ou bidirecionais; (ii) iniciado pelo “chamador”; (iii) possibilita a decomposição/modularização do sistema.
- Desvantagens: (i) dificulta o controle da propagação dos erros, uma vez que a comunicação entre os componentes pode ser relaxada; (ii) quando bidirecional, implica num maior acoplamento entre os módulos.
- Variantes:
  - Sub-rotinas;
  - Orientada a objetos;
  - Cliente-Servidor;
  - *Peer to peer*;
  - Camadas.

## 5. Cliente-Servidor

- Descrição: Os sub-sistemas servidores oferecem os serviços a múltiplas instâncias de sub-sistemas clientes. Os clientes não se comunicam entre si.
- Problema: Sistemas com a necessidade de separação explícita entre as funcionalidades relativas à execução do sistema (servidor) e ao tratamento dos retornos do sistema (cliente). Normalmente é utilizado em aplicações Web, quer seja puro, quer seja em conjunto com o estilo arquitetural em camadas.
- Vantagens: (i) arquiteturas cliente/servidor aumentam a escalabilidade do sistema, uma vez que é possível adicionar servidores gradativamente; (ii) promove transparência de plataforma e de localização entre o cliente e o servidor.
- Desvantagens: (i) não possibilita uma comunicação *peer to peer*.

## 6. Camadas

- Descrição: Uma arquitetura em camadas é organizada hierarquicamente. Cada camada oferece serviço para a camada imediatamente acima e utiliza os serviços oferecidos pela camada imediatamente abaixo. Essa restrição de comunicação pode ser relaxada por questões de desempenho.
- Problema: Um sistema grande que é caracterizado por um conjunto de partes de alto e baixo níveis, onde as partes de nível mais elevado dependem das partes de nível menos elevado.
- Vantagens: (i) baixo acoplamento entre os componentes arquiteturais (melhor *manutenibilidade* e flexibilidade), uma vez que a comunicação é restrita a um dos sentidos; (ii) melhora o entendimento, uma vez que reduz a complexidade do sistema (representação em níveis); (iii) maior grau de reutilização, tendo em vista a maior independência.
- Desvantagens: (i) o desempenho pode ser comprometido, pois pode impor um nível de indireção desnecessário; (ii) nem todos os sistemas são mapeados facilmente em camadas.

## 7. Máquina Virtual

- Descrição: Em uma arquitetura de máquina virtual, o sistema é estruturado de forma a dividir a sua execução em dois passos: (i) interpretação dos comandos; e (ii) execução propriamente dita. Por essa razão, esse estilo arquitetural define um componente responsável unicamente pelo processo de tradução. Esse estilo pode ser representado através do estilo em camadas, onde uma das camadas é responsável pelo processo de tradução.
  - Problema: Necessidade de desconhecer totalmente o ambiente onde o sistema será executado, como por exemplo, máquinas virtuais e/ou sistemas multiplataforma.
  - Vantagens: (i) abstrai o hardware específico ou real que necessita para executar (maior portabilidade e testabilidade).
- Desvantagens: (i) overhead de tradução para a linguagem nativa (menor desempenho).

## 8. Componentes independentes

- Descrição: Uma arquitetura de componentes independentes representa o sistema através de um conjunto de processadores independentes (componentes), que se comunicam através de mensagens. Vale ressaltar que um componente não tem controle sobre o outro.

- Problema: Sistemas modulares, compostos de componentes autônomos, que se comunicam entre si através de troca de mensagens. Nesses sistemas, os componentes podem ser vistos como sub-sistemas, que em outros contextos podem ser utilizados como sistemas particulares.
- Vantagens: (i) alta escalabilidade, como consequência da possibilidade de se adicionar componentes gradativamente ao sistema; (ii) o tempo de mercado é reduzido, uma vez que essas unidades de processamento (os componentes) podem ser reutilizados e/ou desenvolvidos incrementalmente.
- Desvantagens: (i) o fluxo de interação do sistema pode ficar confuso (compromete o entendimento e a manutenibilidade). Para corrigir essa deficiência, costuma-se utilizar esse estilo associado a outro mais restrito, como por exemplo o estilo em camadas.
- Variantes:
  - *Communicating process* – arquitetura orientada a serviço ;
  - Baseado em eventos.

## 9. Heterogêneo

- Descrição: Uma combinação de vários estilos arquiteturais.
- Problema: Situações onde a adoção de um estilo específico não atende todos os requisitos e/ou não resolve o problema satisfatoriamente.
- Vantagens: (i) tenta conciliar as vantagens de vários estilos arquiteturais, amenizando as desvantagens através de um ponto de equilíbrio entre as características de vários estilos.
- Desvantagens: (i) a qualidade de uma arquitetura heterogênea depende muito da experiência do arquiteto.