



**Cláudio Jorge Vieira
Teixeira**

**INFRA-ESTRUTURA PARA
INTEGRADOR DE SERVIÇOS**

PORTAL INTERNET



**Cláudio Jorge Vieira
Teixeira**

**INFRA-ESTRUTURA PARA PORTAL INTERNET
INTEGRADOR DE SERVIÇOS**

Tese apresentada à Universidade de Aveiro para cumprimento dos requisitos necessários à obtenção do grau de Doutor em Engenharia Informática, realizada sob a orientação científica do Dr. Joaquim Sousa Pinto, Professor Auxiliar do Departamento de Electrónica Telecomunicações e Informática da Universidade de Aveiro e do Dr. Joaquim Arnaldo Martins, Professor Catedrático do Departamento de Electrónica Telecomunicações e Informática.

Apoio financeiro da FCT e do FSE no âmbito do III Quadro Comunitário de Apoio.

Dedico este trabalho ao meu pai, Jorge, e às 3 mulheres da minha vida: a minha mãe, Fátima, a minha irmã, Carina, e a Sarah, a futura mãe dos meus filhos.

o júri

Presidente

Doutora Maria Hermínia Deulonder Correia Amado Laurel
Professora Catedrática da Universidade de Aveiro

Doutor Joaquim Arnaldo de Carvalho Martins
Professor Catedrático da Universidade de Aveiro

Doutor Gabriel de Sousa Torcato David
Professor Associado da Faculdade de Engenharia da Universidade do Porto

Doutor Leonel Duarte dos Santos
Professor Auxiliar da Escola de Engenharia da Universidade do Minho

Doutor Joaquim Manuel Henriques de Sousa Pinto
Professor Auxiliar da Universidade de Aveiro

Doutor José Manuel Matos Moreira
Professor Auxiliar Convidado da Universidade de Aveiro

Doutor Rogério Patrício Chagas Nascimento
Professor Adjunto I, do Centro de Ciências Exatas e Tecnologia da
Universidade Federal de Sergipe, Brasil

agradecimentos

A realização deste trabalho foi apenas possível graças à colaboração e apoio de um conjunto alargado de pessoas. A todos estarei para sempre grato.

Agradeço em particular aos meus orientadores, por toda a paciência, aconselhamento e apoio ao longo destes anos. Um agradecimento especial ao Prof. Sousa Pinto, pelas inúmeras conversas e sugestões relacionadas com todas as áreas do nosso interesse, e pela exigência na investigação e no trabalho.

Agradeço à Universidade de Aveiro e ao Instituto de Engenharia Electrónica e Telemática de Aveiro, o acolhimento e disponibilização de recursos para a execução deste trabalho.

Agradeço a todos os docentes do meu tempo de aluno de Engenharia de Computadores e Telemática, pela qualidade e excelência das aulas leccionadas, assim como pela disponibilidade para a participação em actividades extra-curriculares. É também graças a vós que percorro esta etapa.

Agradeço aos meus colegas e amigos de curso e, desde então, de sempre. Sem todos vós, nenhum de nós seria o que é hoje.

Agradeço aos meus amigos de sempre, por entenderem que a não presença não é sinónimo de ausência.

Agradeço aos meus pais todos os esforços que fizeram para que pudesse chegar ao final desta etapa. A eles devo tudo.

Agradeço à minha família todo o encorajamento e apoio incondicional em todas as minhas decisões.

Agradeço à minha namorada toda a paciência e compreensão.

palavras-chave

perfil de utilizador, perfis dinâmicos, estereótipos, portal web, integração de serviços, SOA.

Resumo

Esta tese surge no contexto de sistemas e serviços web. O objectivo é propor uma solução para o problema da integração de informação de diversas fontes, numa plataforma web única, personalizável e adaptável ao utilizador. Nos casos de empresas ou organizações que tenham para diferentes tarefas, diferentes sistemas de informação independentes, o problema da integração de informação surge com a necessidade de integração destes numa única interface disponibilizada aos seus utilizadores.

A integração de serviços numa mesma interface pressupõe que haja comunicação entre um sistema central (que fornece a interface) e os diversos sistemas existentes (que detêm a totalidade – ou parte – da informação a apresentar). Assim, será necessário garantir a identidade do utilizador a cada um dos serviços apresentados, bem como assegurar que cada utilizador tem à sua disposição de forma centralizada, apenas e só a informação e operações a que realmente tem acesso em cada um dos sistemas. Trata-se de uma plataforma que pretende por um lado, fornecer a informação correcta e orientada ao utilizador e, por outro lado, garantir que a organização que suporta o sistema consegue informar e interagir com os seus utilizadores de forma mais eficaz.

O cenário adoptado é a Universidade de Aveiro. Esta pretende disponibilizar uma plataforma electrónica, onde os diferentes interlocutores (alunos, docentes, funcionários, ex-alunos, etc.) possam ter acesso a informação dirigida e orientada aos seus interesses e funções na Universidade.

De modo a que cada utilizador seja realmente visto como um utilizador único, serão estudados e comparados serviços de modelação de utilizador e perfis de utilizador. Será proposto um serviço de modelação de utilizador e uma lógica de criação de perfis de utilizador, distintos do existente no estado de arte. Esta lógica conjuga a personalização da interface por parte do utilizador, com a gestão de operações e definição de políticas de segurança por parte da organização, de forma independente relativamente ao sistema de informação subjacente.

keywords

user profile, dynamic user profile, stereotypes, web portal, services integration, SOA

abstract

This thesis' focus is on web systems and services. Its goal is propose a solution for the multiple information sources integration on a single web platform, customizable and adaptable by the user. In cases where companies and organizations have for different tasks, different information systems, the information integration problem comes with the need of assembly of this disperse information in a single interface for their users.

Single interface service integration requires the communication between a central platform system (that supplies the interface) and the spread systems that hold the hole (or part of the) information to present to the users. Therefore, it will be necessary to guarantee the user's identity to each of the presented services, as well as to ensure that each user will only access the information and functionalities to which he has clearance in those systems. This platform aims at providing on one hand, the proper information for the user, and on the other, guarantee that the organization that supports the system can inform and interact with its users in a more effective manner.

The adopted scenario for testing is the Universidade de Aveiro. The university plans on making available an electronic platform where the different users (students, teachers, non-teaching staff, alumnae, etc.) may have access to information directly guided towards their interests and roles at the University.

In order to have each user seen as a unique user, user modeling servers and user profiles will be studied and compared. It will be proposed a user modeling server and a user profile approach, both different from the state of the art approaches. This approach encompasses the interface customization by the user with operation management and security policy definition by the organization, in an independent logic from the overall information system.

Índice

Índice	1
Lista de acrónimos.....	5
Índice de figuras	7
Índice de tabelas	9
1 Introdução.....	10
1.1 Contextualização.....	10
1.2 Motivação	11
1.3 Objectivos	12
1.4 Principais contribuições.....	14
1.5 Organização da dissertação.....	14
2 Portais web	16
2.1 Introdução	16
2.2 MyEmpresa.....	18
2.3 Vignette.....	18
2.4 Basic Support for Evolutionary Web Portals.....	19
2.5 uPortal.....	21
2.6 Microsoft Office SharePoint Server	24
2.6.1 Arquitectura do sistema	25
2.6.2 O front-end	26
2.6.3 Serviços internos	27
2.7 BEA	28
2.8 TIBCO PortalBuilder.....	30
2.9 Comparação de sistemas.....	30
2.10 Conclusão.....	31
3 Single Sign-On.....	33
3.1 Security Assertion Markup Language	33
3.2 Normas de alto nível.....	36
3.2.1 WS-Federation.....	36
3.2.2 Liberty Alliance.....	38
3.2.3 eXtensible Access Control Markup Language.....	39
3.3 Aplicações SSO	41
3.3.1 Shibboleth.....	41

3.3.2	CardSpace.....	43
3.3.3	Windows Live ID	45
3.3.4	Eduroam	47
4	Perfil de utilizador.....	49
4.1	Noção de perfil de utilizador	49
4.2	Estereótipos	50
4.3	Sistemas de Modelação de Utilizador	52
4.3.1	General User Model System	53
4.3.2	User Modeling Tool	54
4.3.3	TAGUS.....	55
4.3.4	Personis	55
4.3.5	User Modeling Server	57
4.3.6	Generic User Modeling Server for Adaptive Web Systems	60
4.3.7	Comparação dos sistemas de modelação	62
4.4	Armazenamento de Perfis de Utilizador.....	63
4.4.1	Bases de dados relacionais	64
4.4.2	Bases de dados XML nativas	65
4.4.3	Sistemas de ficheiros.....	65
4.4.4	Comparação entre sistemas de armazenamento	66
5	O Portal my.ua.....	68
5.1	Introdução.....	68
5.2	Arquitectura	69
5.3	Modelo de Dados.....	70
5.4	Integração de sistemas e serviços	71
5.4.1	Arquitecturas orientadas a serviços.....	72
5.4.2	A arquitectura proposta	74
5.5	Integração do perfil de utilizador no portal	77
5.5.1	Elementos base	78
5.5.2	Definição de perfil básico	82
5.5.3	Perfil de utilizador.....	83
5.5.4	Alterações ao perfil de utilizador	85
6	Sistema de gestão de perfis	86
6.1	API - Interface de programação.....	87
6.2	Implicações de segurança	90

6.3	Definição do processo formal	91
6.4	Acesso por diferentes aplicações	93
6.5	Comparação com servidores de modelação de utilizadores	95
6.5.1	Semelhanças	95
6.5.2	Diferenças e melhorias	96
6.5.3	Vantagens e limitações	97
6.6	Conclusão.....	98
7	my.ua.pt – O sistema	100
7.1	Leitor de notícias	102
7.2	Informação pessoal dos serviços académicos	103
7.3	Aulas de hoje	104
7.4	Cliente de correio electrónico e agenda	104
7.5	Mensagens gestão documental.....	108
7.6	Sumário de mensagens de contas de correio.....	108
7.7	Agenda pessoal	109
7.8	Informação dos serviços académicos.....	112
7.8.1	Horário lectivo.....	112
7.8.2	Informação de disciplinas.....	112
7.8.3	Informação de turmas	113
7.9	Explorador de ficheiros da sua área pessoal	114
7.10	Editor de páginas HTML da sua área web	116
7.11	Leitor de conteúdos noticiosos preferidos.....	116
7.12	Actualização de dados pessoais no serviço central	117
7.13	Alteração de palavra passe	117
7.14	Alteração da configuração das diversas páginas	118
7.15	Sobre os componentes.....	119
8	Resultados.....	121
8.1	Receptividade e uso – dados estatísticos	122
8.1.1	Visitas	123
8.1.2	Páginas vistas.....	124
8.1.3	Páginas por visita.....	125
8.1.4	Tempo médio por visita.....	126
8.1.5	Páginas mais visitadas	126
8.1.6	Dispersão geográfica	127

8.2	Testes de usabilidade	128
8.3	Gestão de informação armazenada em XML	130
9	Conclusões	134
9.1	Sobre os objectivos do sistema.....	134
9.2	Perspectivas de evolução	136
9.3	Considerações finais	137
10	Referências	140

Lista de acrónimos

AJAX	Asynchronous JavaScript and XML
API	Application Programming Interface
ARCA	Arquivo Central de Conteúdos
BaSeWeP	Basic Support for Evolutionary Web Portals
BPD	Basic Profile Definition
BPEL	Business Process Execution Language
BPM	Business Process Model
CICUA	Centro de Informática e Comunicações
CMS	Content Management System
CORBA	Common Object Request Broker Architecture
CPAD	Customizable Personalized Adaptive Desktop
CRUD	Create, Read, Update and Delete
DLL	Dynamic Link Library
DRH	Divisão de Recursos Humanos
eABC	Arquivo Bibliográfico e Científico da Universidade de Aveiro
Eduroam	Education Roaming
GUMS	General User Model System
GUMSAWS	Generic User Modeling Server for Adaptive Web Systems
HEP	Horizontal Enterprise Portal
HTTP	HyperText Transfer Protocol
ID-FF	Identity Federation Framework
ID-WSF	Identity Web Services Framework
IMAP4	Internet Message Access Protocol 4
IP	Identity Provider
JA-SIG	Java Architectures Special Interest Group
JDBC	Java Database Connectivity
LDAP	Lightweight Directory Access Protocol
NXD	Native XML Database
OASIS	Organization for the Advancement of Structured Information Standards
ODBC	Open Database Connectivity
PACO	Portal Académico

PAP	Ponto de Administração de Política
PDP	Ponto de Decisão de Política
PEP	Ponto de Execução de Política
RCU	Registo Central de Utilizadores
REST	Representational State Transfer
RIA	Rich Interactive Applications
RP	Resource Provider
RSS	RDF Site Summary, Really Simple Syndication
SAC	Serviços Académicos
SAML	Security Assertion Markup Language
SAS	Serviços de Acção Social
SDOC	Serviços de Documentação
SGBD	Sistema de Gestão de Bases de Dados
SOA	Service Oriented Architecture
SOAP	Simple Object Access Protocol
SP	Service Provider
SSO	Single Sign-On
STS	Security Token Service
UA	Universidade de Aveiro
UDDI	Universal Description Discovery and Integration
UMS	User Modeling Server
UMT	User Modeling Tool
UUP	Unified User Profile
VEP	Vertical Enterprise Portal
WAYF	Where Are You From
WRSP	Web Services for Remote Portlets
WSDL	Web Services Description Language
WYSIWIG	What You See Is What You Get
XACML	eXtensible Access Control Markup Language
XEnDB	XML Enabled Database
XHTML	eXtensible Hypertext Markup Language
XML	eXtensible Markup Language

Índice de figuras

Figura 1- Arquitectura geral do uPortal 3.0.....	24
Figura 2 - Diagrama de serviços do SharePoint Portal Server.	26
Figura 3 - Arquitectura do BEA WebLogic Portal (BEA Systems Inc., 2008).....	29
Figura 4 - Arquitectura do BEA WebLogic Platform (BEA Systems Inc., 2008)	29
Figura 5 - Relação entre os diferentes blocos que definem o SAML.....	35
Figura 6 - Diagrama da componentes associados a WS-Federation	36
Figura 7 - Modelo do STS (Goudner et al., 2007).....	37
Figura 8 - Diagrama de funcionamento do XACML	40
Figura 9 - Esquema de funcionamento do Shibboleth.....	42
Figura 10 - Fluxo de troca de mensagens para acesso a recursos protegidos, usando o CardSpace.....	44
Figura 11 - Fluxo de mensagens para permitir o acesso de uma aplicação a dados do utilizador.....	46
Figura 12 - Estrutura geral do Eduroam	48
Figura 13 - Esquema de funcionamento do Personis em cenários multi-aplicação e arquitectura interna do servidor	56
Figura 14 - Arquitectura geral do UMS	58
Figura 15 - Navegação LDAP no UMS (Kobsa & Fink, 2006)	59
Figura 16 - Arquitectura do GUMSAWS.....	61
Figura 17 – Vista prática do portal como integrador de informação.....	69
Figura 18 - Arquitectura do sistema	70
Figura 19 - Serviços de informação existentes na Universidade.....	75
Figura 20 - Diagrama de classes do elemento Operação (parte 1)	79
Figura 21 - Diagrama de classes do elemento Operação (parte 2)	80

Figura 22- Esquema de perfil básico	83
Figura 23 - Esquema completo do perfil de utilizador	84
Figura 24 - Diagrama de classes do sistema de gestão de perfis.....	88
Figura 25 - Métodos do gestor de serviços	89
Figura 26 - Métodos do gestor de operações.....	90
Figura 27 - Classe nível de segurança	91
Figura 28 - Modelo de três camadas aplicado à interacção entre aplicações, sistema de gestão de perfis e sistema anfitrião.	94
Figura 29 - Esquematização dos contextos e componentes (operações) desenvolvidas ...	100
Figura 30 - Aspecto do leitor de notícias integrado	103
Figura 31 - Informação PACO	103
Figura 32 - Informação de aulas para o dia	104
Figura 33 - Aspecto geral do cliente de correio electrónico desenvolvido	107
Figura 34 - Interface agenda do componente de correio electrónico	107
Figura 35 – Componente informativo de mensagens de workflow na gestão documental	108
Figura 36 - Sumário de novas mensagens de correio electrónico	109
Figura 37 - Agenda semanal (vista 1)	110
Figura 38 - Agenda semanal (vista 2)	110
Figura 39 - Agenda semanal (vista 3)	111
Figura 40 - Interface de configuração de agenda	111
Figura 41 - Horário lectivo.....	112
Figura 42 - Informação orgânica sobre uma disciplina.....	113
Figura 43 - Lista de sumários	114
Figura 44 - Lista de alunos de uma turma.....	114
Figura 45 - Interface acesso área pessoal de documentos	115

Figura 46 - Lista de conteúdos noticiosos	116
Figura 47 - Gestão de canais noticiosos	117
Figura 48 – Interface de actualização de dados	117
Figura 49 - Configuração da disposição dos diversos componentes	118
Figura 50 - Rearranjo da página de entrada.....	119
Figura 51- Sistema de informação pós my.ua	121
Figura 52 - Gráfico comparativo do número de visitas nos anos de 2007 e 2008.....	124
Figura 53 - Gráfico comparativo do número de páginas vistas nos anos de 2007 e 2008	125
Figura 54 - Gráfico comparativo de páginas por visita nos anos de 2007 e 2008.....	125
Figura 55 - Gráfico comparativo do tempo médio por visita nos anos de 2007 e 2008....	126
Figura 56 - Páginas mais visitadas e interacções.....	127
Figura 57 - Dispersão geográfica de acessos ao my.ua	128

Índice de tabelas

Tabela 1 - Comparação dos diferentes portais apresentados.....	31
Tabela 2 - Exemplo de um estereótipo do Grundy.....	51
Tabela 3 - Comparação de sistemas de modelação	63
Tabela 4 - Comparação entre os diferentes modelos de armazenamento.....	67
Tabela 5- Principais diferenças e melhorias entre o sistema de gestão de perfis e o estado de arte	97
Tabela 6 - Quadro resumo de componentes e fontes de informação.....	120
Tabela 7 - Tempos de pesquisa (em segundos).....	132

1 Introdução

1.1 Contextualização

É hoje habitual em organizações grandes e complexas, a existência de múltiplos sistemas de informação destinados a suprir as necessidades de um serviço ou sector de actividade. Como as necessidades e muitas vezes os utilizadores dos diversos sistemas são distintos, estes são desenvolvidos com interfaces próprias, mecanismos de controlo de acessos e repositório de dados. Estes múltiplos sistemas comportam-se normalmente como ilhas, não partilhando dados entre si. Além disso, há informação replicada entre todos, como por exemplo os dados pessoais dos seus utilizadores.

Com o avanço das tecnologias da informação, e por pressão dos utilizadores dos diversos sistemas (que têm de manter múltiplas contas para acesso aos diversos sistemas), essas empresas tentam hoje organizar a informação de forma consistente. O próprio paradigma de acesso a esses sistemas mudou nos últimos anos. Até há uns anos, a interface com o utilizador dos diversos sistemas era feita por aplicações próprias. Hoje em dia, a interface universal é o navegador de páginas de internet (Web browser).

Segundo este novo paradigma, e de modo a integrar os diversos serviços numa interface única, tendo a Web como veículo de transporte e disponibilização da informação, surge a noção de “portal web”. Num “portal web”, a informação disponibilizada ao utilizador é baseada na ideia da “informação certa para a audiência certa”. Esta mudança de filosofia faz com que a interface deixe de ser orientada à instituição e passe a ser orientada ao utilizador. Os desafios que se colocam no desenvolvimento de “portais web”, para empresas com múltiplos serviços, são complexos e resultam em tarefas árduas: nem todos os utilizadores têm acesso a todos os serviços da instituição; os diversos utilizadores de um mesmo serviço têm diferentes necessidades e acesso a partes distintas da informação disponibilizada, etc. Também não é financeira nem temporalmente viável parar os sistemas em funcionamento para desenvolver tudo de novo segundo este novo paradigma. É preciso criar “pontes” para interligar todas estas “ilhas” e, se possível, através destas pontes, gerar novos serviços.

1.2 Motivação

O cenário de estudo deste trabalho tem como universo a Universidade de Aveiro (UA), com os seus diversos sistemas de informação e actores.

A informação na UA está dispersa por vários sistemas que se comportam como ilhas. Como exemplos, podemos referir os Serviços Académicos (SAC), a Divisão de Recursos Humanos (DRH), os Serviços de Acção Social (SAS), o Centro de Informática e Comunicações (CICUA) e os Serviços de Documentação (SDOC), de entre os mais importantes.

Estes “sistemas ilha” sabem muito pouco sobre os demais, têm o seu grupo de utilizadores, as suas regras de acesso e lógicas de negócio próprias. Atendendo a que existe um universo global de utilizadores dos diferentes sistemas (pessoas ligadas por um determinado vínculo à UA – sejam eles alunos, bolsiros, docentes ou funcionários) seria lógico que os diferentes sistemas usassem os utilizadores do universo geral de utilizadores da UA. Ora, acontece que este não é o caso. Cada sistema gere os seus utilizadores, sem qualquer ligação com os demais. Numa primeira análise, trata-se de um procedimento que garante a total independência dos sistemas. Contudo, fazendo uma análise mais aprofundada, conclui-se que há tarefas que são repetidas em cada um dos sistemas e que por ventura poderiam estar incumbidas a um único sistema. A parte de gestão de utilizadores, mecanismos de armazenamento e recuperação de palavras passe, regras de palavras passe seguras, entre outras, é repetida podendo ser tarefa de um sistema transversal a todos os outros.

Este sistema transversal não poderá, em nenhuma circunstância, imiscuir-se nas regras de negócio próprias dos sistemas já existentes. A sua única função é servir de repositório de informação comum a todos os sistemas e garantir a autenticação de utilizadores. Será da responsabilidade de cada sistema garantir a autorização de utilizadores de acordo com as regras de negócio implementadas.

A criação de um sistema de identificação central nestas condições é um marco importante na gestão da informação de utilizadores da UA. Estando a universidade a iniciar este passo, surge a oportunidade de construir pontes entre os diversos sistemas, agregando informação de fontes distintas, para poder inovar na forma de a representar. Com estas pontes, poderão

ser criados serviços de valor adicional ao sistema, resultado das sinergias entre sistemas existentes.

A arquitectura para portal internet integrador de serviços aqui proposta é um exemplo de interligação de sistemas, sendo o portal em si visto como uma “montra” de informação para o utilizador.

1.3 Objectivos

Esta tese apresenta uma arquitectura de portal web, com integração de múltiplos serviços numa única interface centrada no utilizador. Serão simultaneamente estudadas as consequências que uma arquitectura como a proposta acarreta no armazenamento e disponibilização da informação dos diversos serviços.

O caso de estudo utilizado é um cenário institucional, mas a lógica inerente à arquitectura é passível de ser facilmente integrada em cenários empresariais.

Como se pretende uma interface centrada no utilizador, isso significa que será possível a cada utilizador dispor na sua área de trabalho as interfaces “cedidas” por cada um dos serviços em que tenha autorização de acesso. Pretende-se também que a informação representada por cada serviço na interface do utilizador seja configurável. Concretizando, pretende-se que cada utilizador, em função das suas permissões num dado serviço, tenha a possibilidade de configurar o que quer ver e como quer ver.

Com a abordagem escolhida, que passa pela integração de serviços e não pela sua substituição, nenhum dos actuais serviços perde a sua identidade própria; criam-se canais de comunicação com o serviço, o que faz com que este se possa abrir para o exterior e decidir, em função do serviço ou do utilizador que acede, qual a informação que lhe é relevante. É também uma abordagem incremental, pois permite que os serviços aumentem a quantidade de informação disponibilizada em função dos seus meios e interesses, ou que ao longo do tempo disponibilizem novas funcionalidades, não sendo necessário que todo o esforço de integração se faça de uma só vez.

É também objectivo desta tese avaliar, genericamente, qual o melhor meio de comunicação entre os serviços e o portal e entre serviços. Serão analisados os aspectos respeitantes quer à conectividade entre sistemas, quer à segurança na transmissão dos dados.

A informação de utilizadores e serviços será organizada em perfis.

Às definições referentes ao modo de acesso a um dado serviço, bem como qual o tipo de informação que este pode disponibilizar, designamos por “perfil de serviço”.

O conjunto de serviços a que cada utilizador tem acesso, juntamente com a informação que este programou para receber dos serviços e ainda o local onde os blocos de informação dos serviços são colocados no seu ecrã de trabalho, constitui a parte transaccional do “perfil de utilizador”.

Para além do acesso e organização espacial dos serviços no ecrã do utilizador, pretende-se também estudar e avaliar qual a melhor forma de suportar os dados dos diversos perfis. É expectável que um portal deste tipo tenha uma utilização intensiva por parte dos utilizadores. Colocam-se, entre outras, as seguintes questões:

- Qual o impacto do número de utilizadores no desempenho do portal?
- Qual o impacto desta metodologia de acesso nos diversos serviços?
- Qual o modo mais seguro de comunicar entre serviços?
- Como permitir que um utilizador, uma vez ligado ao portal, possa passar para um dos subsistemas já existentes sem que seja necessária uma nova identificação (troca de credenciais entre sistemas / identificação única para os diversos sistemas)?

Para responder a estas perguntas serão estudados e testados os paradigmas de acesso aos dados por bases de dados relacionais, bases de dados não relacionais (como é o caso de bases de dados *eXtensible Markup Language* (XML)) e bases de dados orientados ao sistema de ficheiros. Embora as primeiras tenham a primazia da tecnologia existente, as restantes abordagens poderão também ser interessantes, sendo por isso necessário estudar e avaliar qual a que melhor se adequa em diferentes cenários.

Como meio de comunicação entre sistemas, será proposto um modelo *Service Oriented Architecture* (SOA) baseado em *web services* (Booth et al., 2004; Gailey, 2003). Pela sua natureza, o uso de *web services* permite a comunicação com sistemas que estejam protegidos com firewalls ou zonas desmilitarizadas. Uma vez que os *web services* utilizam envelopes *Simple Object Access Protocol* (SOAP) (Gudgin et al., 2007) e comunicam via *HyperText Transfer Protocol* (HTTP), as questões de comunicação segura serão também tidas em consideração.

Para que os diferentes sistemas possam comunicar com segurança e identificar univocamente o utilizador, será necessário implementar um mecanismo de *Single Sign-On* (SSO), em que as credenciais do utilizador são comunicadas entre os diversos sistemas, para que estes o possam identificar nas diferentes operações. Este mecanismo será testado na passagem de credenciais entre diferentes sistemas, de modo que o utilizador possa aceder aos serviços web próprios de cada sistema mantendo sempre a sua autenticação.

1.4 Principais contribuições

As principais contribuições desta tese no problema da plataforma integradora de serviços são:

- Uma arquitectura de portal web altamente configurável, onde a informação, componentes e operações constantes em cada página, dependem das permissões constantes no perfil do utilizador que aceder naquele instante;
- Um sistema de modelação de perfis de utilizador, totalmente independente das aplicações que o possam usar, cuja principal função é personalizar o acesso a serviços dentro de um ambiente corporativo, baseado em cargos e funções dinâmicas;
- Uma lógica de perfis de utilizador baseada em perfis base estruturados de forma hierárquica;
- O conceito de processo formal, mediante o qual são definidas regras de atribuição e validação de acesso a operações no perfil de utilizador;
- Um estudo sobre a performance da pesquisa de informação estruturada em documentos XML, armazenados em diversos repositórios e pesquisada de diversas formas.

1.5 Organização da dissertação

Esta tese é composta por nove capítulos, sendo este o primeiro, onde se contextualiza a problemática, motivação e objectivos da investigação.

Os capítulos dois, três e quatro analisam o estado da arte, sendo inteiramente dedicados ao estudo de tecnologias de suporte e estudo de sistemas análogos ao que se pretende desenvolver. Concretamente, o capítulo dois apresenta diversas soluções de portais web, procurando explicar cada uma das diferentes abordagens seguidas. No final do capítulo é

feita uma comparação de forma sucinta entre todos os sistemas apresentados. O capítulo três analisa o estado de arte de protocolos e aplicações de SSO. No capítulo quatro é apresentado o estado de arte de perfil de utilizador e são discutidos os sistemas de modelação de utilizadores com maior relevo. Depois de analisados de forma individual, os sistemas de modelação são também comparados. Este capítulo termina com a discussão e comparação das diferentes alternativas ao armazenamento de perfis de utilizador.

Os capítulos cinco, seis e sete são dedicados à apresentação da solução proposta para o problema e perguntas de partida. Concretamente, o capítulo cinco apresenta o sistema desenvolvido, o portal de nome “my.ua”. Neste capítulo são apresentados e discutidos aspectos relacionados com a arquitectura, modelo de dados e integração de sistemas, serviços e perfil de utilizador. No capítulo seis é apresentado o sistema de gestão de perfis de utilizadores. São apresentados detalhes de segurança e é discutida a forma de integração com o sistema da Universidade. Além disso, este sistema é também comparado com os servidores de modelação de utilizadores apresentados no capítulo 4. O capítulo 7 apresenta o sistema my.ua, do ponto de vista dos componentes desenvolvidos para suportar e integrar as diversas fontes de informação.

O capítulo oito apresenta os resultados de utilização e de usabilidade do sistema, juntamente com um estudo, relevante para o sistema, sobre a gestão da informação armazenada em XML.

O capítulo nove apresenta as conclusões e considerações finais sobre toda a pesquisa e trabalho desenvolvido.

2 Portais web

2.1 Introdução

Os portais web estão cada vez mais presentes na nossa sociedade. Através deles, instituições e empresas disponibilizam informação através de uma interface simples para o utilizador. Os conteúdos disponibilizados nos portais variam de acordo com as especificidades da empresa/instituição.

Antes de avançar numa descrição mais detalhada acerca do que são e para que servem os portais web, é aqui necessário ressaltar a distinção entre portais web e sistemas de gestão de conteúdos – *Content Management System* (CMS). Os CMSs são utilizados para a criação de sistemas que permitam aos utilizadores a partilha de documentos e conteúdos utilizando um serviço central (Williams, 2008). Em (Bergstedt, Wiegrefe, Wittmann, & Moller, 2003) são enumeradas algumas das características que um CMS deverá possuir, destacando-se a gestão de conteúdos, fluxos e utilizadores. Os portais web são sistemas que permitem o acesso consolidado e personalizado a uma série de serviços e informação, incluindo informação institucional ou de alguma forma relacionada com a instituição. O conceito “portal web” tem sido sujeito a várias definições. Zirpins (C. Zirpins, 2001) define-o como o ponto de partida ou entrada para a navegação na Internet. Tsai (C. Tsai, 2005) sumaria algumas das diferentes definições de portal: “local onde os utilizadores iniciam a navegação (...) [e que] disponibiliza motores de pesquisa, email (...) e outros serviços” e “ponto único de acesso a informação” estão entre as definições mais recorrentes.

A personalização do sistema web pode estar presente em ambos os sistemas. Ainda assim, é mais utilizada nos sistemas de portal, para permitir que o utilizador personalize a sua interface e filtre a informação da instituição de forma a corresponder às suas exigências/necessidades, numa visão de portal centrado no utilizador (Smith, 2004). Em Portugal, como exemplos de portais de empresas temos a TMN (TMN, 2005) e a Vodafone (Vodafone, 2005), entre outros.

Como sistemas de criação de portais, entre outros, encontram-se o Vignette (Vignette, 2008), o BaSeWeP (Billig, Gottschick, & Sandkuhl, 2005), o SharePoint Portal da

Microsoft (Microsoft Corporation, 2007), o BEA (BEA Systems Inc., 2008) e o TIBCO Portal (TIBCO, 2008b).

Os portais web podem ser divididos em dois grupos em relação ao seu objectivo (Strauss, 2002): Portais Empresariais Horizontais (*Horizontal Enterprise Portals* - HEPs) e Portais Empresariais Verticais (*Vertical Enterprise Portals* – VEPs).

Enquanto um HEP normalmente é de acesso público e fornece acesso a um vasto número de serviços espalhados na rede, um VEP é um portal de acesso controlado, onde os utilizadores são autenticados na organização. Um VEP deverá ter os mesmos tipos de informação do HEP e mais: informação da organização correlacionada com o utilizador autenticado. É possível ao sistema fornecer este tipo de informação ao utilizador porque a organização sabe quais as actividades ou papéis que este desempenha naquele momento. Assim sendo, a experiência de utilização do portal VEP é mais rica, uma vez que o portal mostra uma vista personalizada da organização.

Se a um VEP se adicionar a possibilidade do utilizador personalizar e adaptar a interface do portal à sua vontade, então o resultado é um ambiente de trabalho personalizado e adaptável (*Customizable Personalized Adaptive Desktop* – CPAD). Num CPAD cada utilizador tem uma interface feita à sua medida, visto que, no limite, dois utilizadores serão sempre utilizadores diferentes (com permissões e gostos distintos).

Como qualquer produto, os portais web evoluem. Neste momento podem ser consideradas três gerações de portais web (Bellas, 2004; Daigle & Cuocco, 2002): a primeira geração tem uma arquitectura tipicamente monolítica, assemelhando-se aos portais públicos, com a ênfase na apresentação de conteúdos; a segunda geração evoluiu para uma arquitectura de *micro-kernel* com conteúdos personalizáveis independentes do motor do portal. Esta abordagem é mais robusta, com óbvios benefícios para o desenvolvimento, manutenção e reusabilidade. Esta geração tem em mente processos institucionais, integração de serviços e informação de formas inovadores, dando origem a serviços de valor acrescentado, uma vez que o ganho da conjugação da informação de diversas fontes é superior à soma dos ganhos individuais das fontes de informação de cada serviço. Finalmente, os portais de terceira geração que, tal como nos telefones, irão integrar múltiplos tipos de dados e formas de acesso, nomeadamente, vídeo-conferência, palestras, registos de voz, ..., bem como suportar um número elevado de dispositivos cliente.

2.2 MyEmpresa

Os portais myEmpresa são cada vez mais comuns. Nestes, o utilizador tem ao seu dispor um conjunto de informações que lhe são relevantes, sejam serviços subscritos, sejam dados pessoais de contratos e facturação.

Trata-se na maioria dos casos de VEPs, visto que a informação neles contida é particular/confidencial, direccionada ao negócio. São exemplo portais como MyTmn (TMN, 2005), MyVodafone (Vodafone, 2005), entre outros. Uma das vantagens destes portais é a virtualização da maioria dos serviços de atendimento ao cliente, seja para pedidos/revogações de serviços, seja para simples consultas de facturação ou suporte. Estes portais estão ao dispor do cliente a qualquer hora, e permitem à empresa poupar em recursos humanos ao nível do atendimento e suporte ao cliente, ao mesmo tempo que fornecem uma imagem de modernidade. A personalização do serviço que alguns portais disponibilizam permite uma maior aproximação ao utilizador, na esperança de aumentar o retorno por parte deste.

2.3 Vignette

Trata-se de uma plataforma comercial com tecnologia para construção de CMS, portais Web e gestão documental, entre outros (Vignette, 2008).

Na parte de portal, disponibilizam dois produtos complementares: o *Vignette Portal* e o *Vignette Builder*. O *Vignette Portal* pode integrar outros produtos da família Vignette, como componentes para colaboração online e/ou comunidades ou grupos de utilizadores. Todos os componentes usados no portal (*portlets*) podem ser consumidores ou fornecedores de eventos na página onde são incorporados. Por exemplo, numa página que contenha um componente de selecção de cidades e outro de previsão de meteorológica de uma determinada cidade, a selecção de uma cidade no selector pode levar à alteração da localização da previsão meteorológica.

Um dos aspectos a ter em consideração neste sistema é a sua compatibilidade com componentes definidos via *Web Services for Remote Portlets* (WSRP (Kropp, Leue, & Thompson, 2003)). Os WSRP podem ser entendidos como *web-services* que, além de informação, enviam também apresentação. A informação, assim como a lógica de acesso e uso da mesma, é encapsulada em porções de *eXtensible Hypertext Markup Language*

(XHTML) que é enviada do fornecedor do serviço para o consumidor. O consumidor por sua vez, está apto a perceber a codificação extra no XHTML e a adicionar esta “peça” ao seu portal.

O *Vignette Builder* é um produto de desenvolvimento de componentes para o portal. Permite que mesmo utilizadores sem experiência como programadores construam novos componentes, através de uma aplicação que lhes permite “arrastar e largar” componentes HTML, tais como botões, caixas de texto, tabelas ou colunas, numa interface *What You See Is What You Get* (WYSIWIG (Brad, 1998)). Esta interface permite a criação de *mashups* para uso no portal. Por *mashup*, entende-se a junção de fontes de dados distintas que permitem a criação de novas formas de ver e tratar a informação (Zou & Pavlovski, 2007). Ainda no campo de interligação de fontes de dados, esta ferramenta automatiza a ligação a fontes de dados externas – como Lotus Domino (IBM, 2008), Siebel e People Soft (adquiridas pela Oracle em 2005), SAP (SAP, 2008), serviços web baseados em SOAP e expressos usando Web Services Description Language (WSDL) (Christensen, Curbera, Meredith, & Weerawarana, 2001), e bases de dados acessíveis via *Open Database Connectivity* (ODBC) e *Java Database Connectivity* (JDBC - (Sun microsystems, 2008)) – garantido deste modo comunicação com praticamente todos os sistemas de gestão de bases de dados (SGBDs) como é o caso de Oracle, DB2, Sybase, Microsoft SQL Server e MySQL.

Da sua lista de clientes fazem parte a Universidade de Miami, Universidade da Califórnia, Universidade de Utah, Bank One, Barclays Gold, Hewlett-Packard, Motorola, Sun Microsystems, Agilent, o Serviço de Correios dos Estados Unidos, entre muitos outros.

2.4 Basic Support for Evolutionary Web Portals

O *Basic Support for Evolutionary Web Portals* (BaSeWeP) é uma plataforma para construção de portais (Billig et al., 2005). Esta plataforma iniciou em produção em 1999, como plataforma de suporte a vários projectos de investigação, fornecendo-lhes diferentes tipos de conteúdos estruturados.

Cada tipo de conteúdo é descrito num esquema composto por:

- referências conceptuais, que categorizam os conteúdos para a web semântica;
- propriedades para descrever os conteúdos usando tipos de dados estruturados;
- referências a recursos existentes no sistema para melhor descrever os conteúdos oferecidos.

A especificação de conteúdos e das regras de autorização de acesso aos mesmos é definida em ficheiros XML. A especificação dos conteúdos é definida pelos seguintes itens: esquema do conteúdo, regras de segurança, regras de mapeamento e finalmente de modelos de apresentação. As regras de mapeamento permitem alguma flexibilidade ao ligar as propriedades do conteúdo a elementos de uma base de dados, garantindo uma certa independência entre conteúdos e dados. Os modelos de apresentação geram a interface gráfica de cada componente, permitindo desta forma adaptar o formato visual dos conteúdos às necessidades de imagem do cliente do portal.

Inicialmente, a construção de um portal envolvia três fases: criação de conteúdos, definição de regras de validação e armazenamento.

A fase de navegação pelo portal englobava também três fases: navegação no espaço de conteúdos, pesquisa específica de conteúdos e, finalmente apresentação de conteúdos.

Esta plataforma sofreu uma série de passos evolutivos que, por sua vez, resultaram no produto comercial VEDA. A versão deste sistema em 2005 (volvidos seis anos do início de uso do sistema) apresenta uma série de alterações praticamente a todos os níveis. A gestão do portal é toda baseada na web, com uma interface de gestão que permite definir todos os conteúdos, a rede semântica, segurança e apresentação do portal. Internamente, todo o código foi internacionalizado e permite acesso multilingue. A arquitectura do portal foi também alterada, sendo a parte semântica da gestão dos conteúdos e conhecimento colocada numa camada própria. No início, as operações sobre conteúdos e conhecimento estavam confinadas ao uso pelo portal. Com a criação da referida camada, o acesso e gestão da base de conhecimento pode ser usada por outros sistemas, através do uso de *web services*. Finalmente, a classificação semântica de conteúdos é parcialmente automatizada, graças à integração de métodos automáticos de classificação.

Do ponto de vista de camadas, o sistema deixa de ser um sistema em 3 camadas para passar para um sistema de quatro camadas:

- armazenamento/bases de dados,
- camada de acesso a dados e a sistemas legados (com as regras de negócio de toda a empresa),
- camada de gestão de conhecimento para uso pelo portal e, finalmente
- camada de interfaces de acesso pelo utilizador.

Apesar da filosofia subjacente ao modelo de informação e da arquitectura do sistema (modelo em quatro camadas) serem interessantes do ponto de vista de plataforma, tanto quanto foi possível apurar, não existem referências ao mesmo após 2005, seja em termos de clientes que usem o sistema, seja em termos de desenvolvimentos da plataforma.

2.5 uPortal

Uma das principais iniciativas de Portais Institucionais é o uPortal, um projecto de código fonte aberto, baseado em JAVA, em desenvolvimento pelo *Java Architectures Special Interest Group* (JA-SIG) – (JA-SIG, 2006). O objectivo deste projecto é desenvolver uma plataforma comum para portais web em universidades. A principal vantagem deste projecto é tratar-se de uma iniciativa inter-universitária. O portal está, actualmente, em uso em várias instituições de educação superior espalhadas pelo mundo entre as quais: Brooklyn College, California Department of Education, Edinburgh University, Plymouth State University, University of British Columbia, University of Hong Kong. A lista completa pode ser acedida em <http://www.uportal.org/who-prod.html>.

O uPortal é dividido em três sistemas distintos:

- a plataforma de acesso a conteúdos (o portal em si),
- *portlets* e
- canais (*Portal channels*), que comportam informação e apresentação em conjunto (Chaoying, Liz, Miltos, & Gill, 2006).

A plataforma de acesso é responsável por gerir e garantir a persistência de informação respeitante a configurações gerais do portal, personalizações dos utilizadores, gestão de autenticação, etc.

Os canais base do sistema são:

- Imagem,
- Página Embebida (*Inline Frame*),
- RSS (*Really Simple Syndication*),
- Web Proxy,
- Transformação de XML Simples,
- Proxy de canal remoto,
- Java applet e
- Feito à medida (*Custom*).

O canal “Imagem” permite mostrar uma imagem na página. A sua utilidade é apenas mostrar a imagem.

O canal “Página Embebida” permite ao portal servir de moldura para outra página que é indicada como parâmetro.

O canal “RSS” permite mostrar a informação contida num canal RSS (por exemplo: notícias ou actualizações de blogues) que é indicado como parâmetro.

O canal “Web Proxy” trata de converter páginas e/ou aplicações web existentes noutros servidores para formato XML que depois serve para alimentar um canal do portal. Permite ainda a portabilidade de aplicações já existentes na organização que está a lançar o portal, permitindo a sua integração neste. Uma vez que o servidor original da aplicação e o servidor do portal não são necessariamente o mesmo, significa que, por um lado, a aplicação pode continuar a ser utilizada como até então e, ao mesmo tempo, permite que o eventual processamento mais intensivo por parte desta não interfira com o comportamento do portal. O processo de conversão para XML do HTML gerado pela aplicação nem sempre é uma tarefa simples e exige HTML correcto, assim como formas de tratar eventos e métodos JavaScript da aplicação original. Embora se trate de um processo complexo, tal conversão é possível.

O canal “Transformação de XML simples” utiliza como canal de informação um XML e uma folha de estilos XSLT, ambos passados como parâmetros.

O canal “Proxy de canal remoto” permite usar canais interactivos provenientes de outros portais. Uma das vantagens mais evidentes desta utilização é possibilitar o uso em vários

portais do mesmo canal, ao mesmo tempo que a manutenção do canal é feita apenas num portal (no portal de origem). Além disso, o processamento do canal é também remoto, pelo que, à semelhança do Web Proxy, não interfere com a disponibilidade do portal em si.

O canal “Java applet” permite usar applets Java no portal.

O canal “Feito à medida” ou canal nativo é desenvolvido e alojado pelo próprio portal. Escrito em Java, implementa interfaces específicas para serem executados pelo portal. Pode servir de agregadores de informação disponibilizada por outros canais ou usar recursos associados ao sistema de informação em que o portal está inserido. Trata-se do tipo de canal mais usado pelo portal. Componentes como o cabeçalho no topo das páginas ou o de login/password, são exemplos de canais feitos à medida.

Com a versão mais recente – uPortal 3 – estes canais passam a ser visualizados via *portlets*, após serem transformados por um adaptador específico em cada caso.

Além dos canais base, há alguns que podem ser instalados, como um canal SQL que permite executar comandos directamente sobre uma fonte de dados.

Os *portlets* obedecem à norma JSR-168 Java e, na versão 2.6 do portal, são compatíveis com WSRP. Na versão 3, a compatibilidade está posta em causa devido a questões internas relacionadas com o motor usado para gestão dos *portlets*: o WSRP é apenas suportado na versão 1.0 desse motor – Pluto – e não é suportado na versão 1.1. Espera-se no entanto o seu suporte na versão 2.0 (Apache Software Foundation, 2008).

Até à versão 2.6 do portal, a ênfase dos conteúdos era colocado nos canais, sendo os *portlets* vistos como um canal específico; na versão 3.0 há uma mudança de filosofia, sendo os *portlets* o cerne da visualização e os canais adaptados a *portlets*.

A Figura 1 representa a arquitectura geral do uPortal (Chaoying et al., 2006). Os objectos dentro da caixa Pluto representam *portlets* geridos pelo Pluto. Estes podem ligar-se directamente à informação externa do portal, ou podem usar adaptadores que fazem a interface dos antigos canais para as *portlets*. Os canais, por sua vez, ligam à informação em causa.

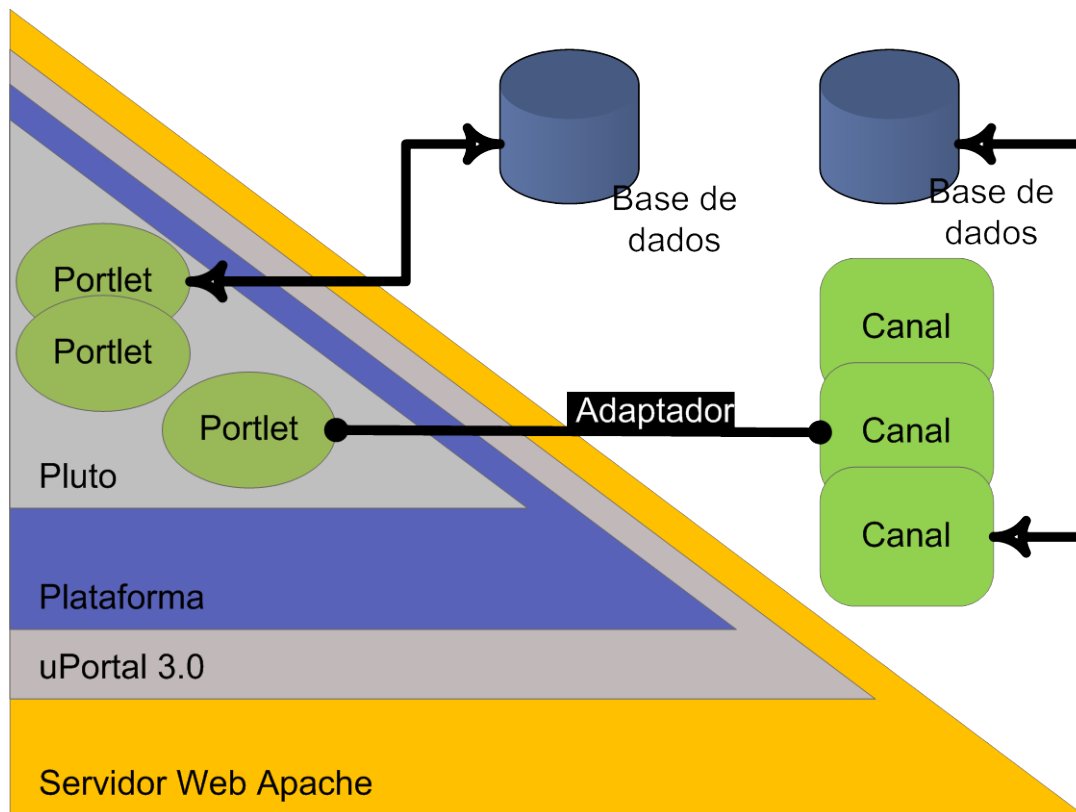


Figura 1- Arquitectura geral do uPortal 3.0

2.6 Microsoft Office SharePoint Server

Esta é a plataforma empresarial da Microsoft para construção de portais web. Sendo um produto Microsoft, integra-se de forma transparente com toda a família de produtos Microsoft Office. Segundo a Microsoft, esta plataforma permite trabalho cooperativo, criação de portais de empresa e de páginas pessoais de utilizadores associadas ao portal, pesquisa, gestão documental, gestão de fluxos de informação e gestão de informação, entre outros, tudo adaptado às funções atribuídas ao utilizador (Microsoft Corporation, 2007).

Uma vez que algumas das suas características não estão directamente ligadas ao âmbito do trabalho em causa, serão explicadas apenas aquelas que são relevantes. A saber:

- páginas pessoais,
- espaços de trabalho,
- pesquisa integrada e
- gestão do sistema.

O portal empresarial pode conter **páginas pessoais** de utilizadores individuais ou grupos de trabalho. Nestas, cada utilizador pode agregar a informação que pretende que lhe seja mostrada, como “documentos/informação para mim”, “documentos/informação da minha autoria” e “informação acerca de mim”. O acesso a esta informação pode ser parcialmente controlado pelo utilizador, distinguindo o que é público, do que é privado e do que é acessível a grupos e/ou utilizadores individuais.

O portal permite a criação de **espaços de trabalho** onde grupos de utilizadores podem trabalhar colaborativamente em documentos (normalmente documentos da família Office).

A **pesquisa integrada** com o portal permite ao utilizador pesquisar informação agregada da empresa, páginas web previamente indexadas, pastas públicas de servidores Microsoft de correio electrónico (servidores Exchange). Esta pesquisa tem em consideração as permissões de acesso do utilizador que pesquisa, de modo a evitar acessos não autorizados.

Toda a **gestão do sistema** pode ser efectuada online, utilizando um browser web para o efeito.

2.6.1 Arquitectura do sistema

O SharePoint Portal Server é composto por vários serviços que podem estar instalados em vários servidores. A Figura 2 representa os canais lógicos de comunicação existente entre os vários serviços desempenhados pelo servidor (English & Microsoft SharePoint Teams, 2004).

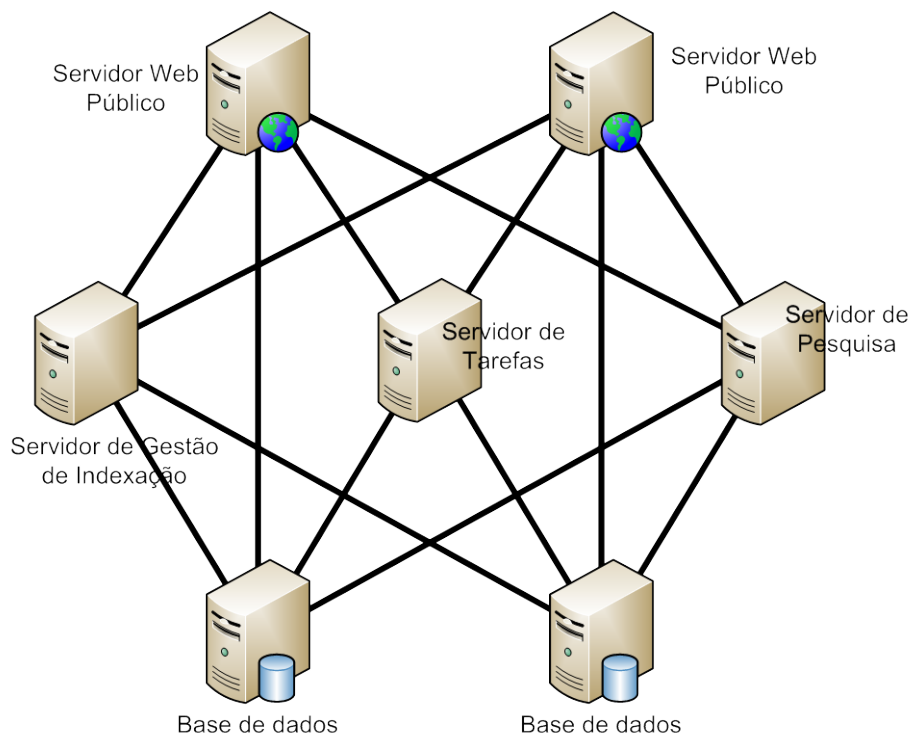


Figura 2 - Diagrama de serviços do SharePoint Portal Server.

O facto dos diferentes serviços serem logicamente distintos é, por si só, um compromisso de escalabilidade, possibilitando uma rápida transição entre um único servidor e vários servidores, desempenhando em paralelo as mesmas funções ou desempenhando funções distintas. De acordo com as necessidades do momento, poderá ser necessário escalar o sistema para uma configuração com um conjunto de servidores web visíveis do exterior (front-end), servidores dedicados a tarefas de tratamento de informação (servidores de indexação, pesquisa e de outras tarefas) e até um conjunto de servidores de base de dados.

A instalação normal deste sistema implica três servidores: um servidor de base de dados, um servidor para as restantes funções (servidores instalados com produtos Microsoft) e, finalmente, um terceiro servidor para funções de controlador de directório (*Active Directory*) e gestão de utilizadores e grupos da empresa.

2.6.2 O front-end

Do ponto de vista de gestão e implementação, as páginas do portal são compostas por *webparts*. Uma *webpart* é conceptualmente semelhante ao conjunto canal e *portlet* da plataforma uPortal: é responsável pela interface e apresentação de dados previamente recolhidos e tratados (pela *webpart*). Além disso, permite que várias *webparts*

comuniquem entre si, ora como fornecedores de informação, ora como consumidores de informação. Um exemplo desta particularidade é, por exemplo, a existência de vários componentes de pesquisa a comunicar com um único componente que mostra os resultados (cada pesquisa poderia ser efectuada em bases de informação distintas).

As regras de segurança estão visíveis a todos os níveis: desde o portal, passando pela página até à *webpart*, a segurança é definida através de audiências. A noção de audiência está relacionada com as funções, cargos e permissões que os utilizadores têm na empresa (atribuídos ao nível do controlador de directório). A um determinado conteúdo apenas podem aceder utilizadores autorizados.

2.6.3 Serviços internos

Como ilustrado na Figura 2, o sistema possui vários serviços internos:

- Servidor de indexação,
- Servidor de tarefas,
- Servidor de pesquisa e
- Servidor de dados.

O servidor de indexação é responsável por garantir o aprovisionamento da informação existente no portal e em fontes externas a este que sejam definidas como fontes de dados (páginas web fora do portal, ficheiros RSS, etc). Os dados recolhidos são tratados e processados para permitir pesquisas rápidas no sistema.

O servidor de tarefas é responsável por todas as tarefas que sejam agendadas. Entre outras, é possível definir os momentos para cópias de segurança da informação do portal ou a indicação do momento em que o serviço de indexação deve iniciar uma busca de novos conteúdos em serviços externos.

O servidor de pesquisa está intrinsecamente ligado ao serviço de indexação, visto que a pesquisa é feita prioritariamente sobre os dados previamente indexados.

O servidor de dados é usado de forma intensiva pelo SharePoint Portal Server. Toda a informação, indexação e tarefas são armazenadas em bases de dados.

Estes serviços podem, todos eles, existir num único servidor, ou poderão ter configurações específicas, com cada serviço a usar um ou mais servidores, se necessário, o que confere a este sistema um elevado grau de escalabilidade.

2.7 BEA

A empresa BEA (BEA Systems Inc., 2008) foi adquirida pela Oracle em Janeiro de 2008. Veio, neste capítulo, suplantando as lacunas da Oracle no domínio de plataformas web para construção de portais web e gestão de conteúdos. A título de curiosidade, o nome BEA resulta das iniciais dos três fundadores da empresa (Bill Coleman, Ed Scott and Alfred Chuang).

Da gama de produtos BEA destaca-se a gama WebLogic:

- WebLogic Server,
- WebLogic Portal,
- WebLogic Integration e
- WebLogic Platform.

O *WebLogic Server* é a base da plataforma, com uma arquitectura SOA e baseado em Java Enterprise Edition (Java EE) 5. Trata-se de uma plataforma escalável e de administração via interface online.

O *WebLogic Portal* permite a disponibilização de informação e conteúdos a todos os parceiros de negócio (funcionários, empresas e clientes), através do perfil de utilizador unificado (*Unified User Profile* – UUP). A informação dos utilizadores pode ser extraída de fontes de dados externas, incluindo via *Lightweight Directory Access Protocol* (LDAP).

A interface de gestão (também ela online) permite construir páginas e conteúdos através do processo de “arrastar e largar” (*drag and drop*). Toda a plataforma obedece a normas na comunicação entre componentes, suportando entre outros, WSRP, *Asynchronous JavaScript and XML* (AJAX) e RSS. Permite, ainda, a comunicação entre os diferentes *portlets* das páginas no lado do cliente, diminuindo as chamadas ao servidor neste processo.

A Figura 3 apresenta a arquitectura do BEA WebPortal.

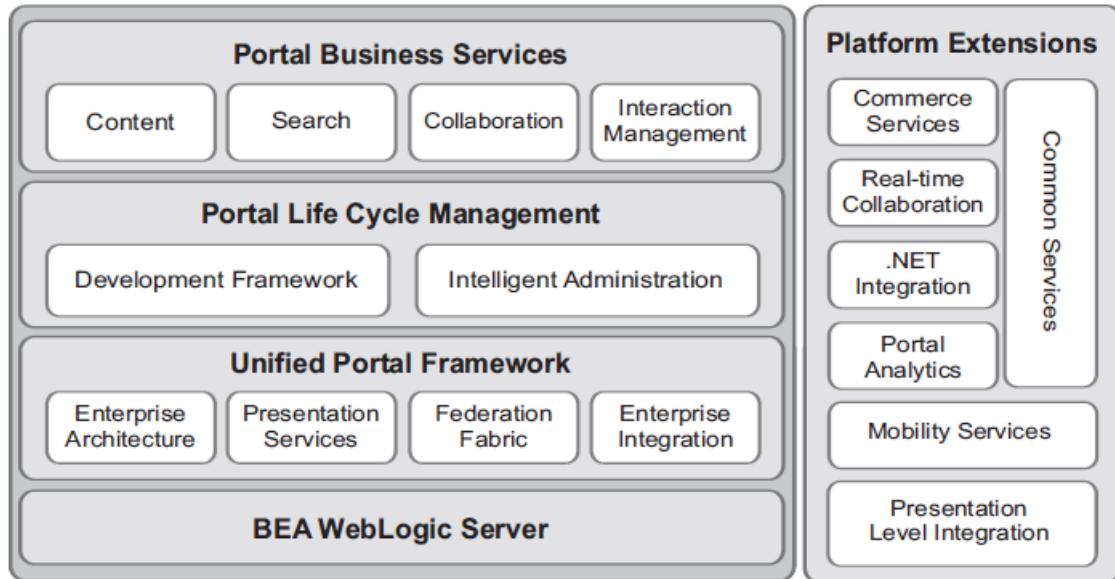


Figura 3 - Arquitetura do BEA WebLogic Portal (BEA Systems Inc., 2008)

O *WebLogic Integration* permite integrar informação da empresa, existente em sistemas legados ou de outros fornecedores na plataforma WebLogic.

O *WebLogic Platform* integra a componente de servidor, de portal, de integração e ainda uma aplicação para desenvolvimento baseada no Eclipse (BEA Workshop for WebLogic).

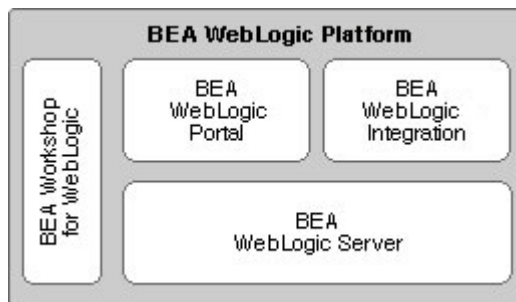


Figura 4 - Arquitetura do BEA WebLogic Platform (BEA Systems Inc., 2008)

A utilização de normas a todos os níveis permite a fácil adoção e integração de componentes e sistemas de outros fornecedores. O desenvolvimento modular de toda a arquitectura permite que cada módulo possa ser modificado para suportar as necessidades específicas de uma empresa em particular.

Uma vez que todo o sistema tem por base Java, este pode ser instalado em plataformas de virtualização tais como o servidor VMware ESX (VMware, 2008).

Da sua carteira de clientes destacam-se o Ministério da Justiça Finlandês, Alibris, T-Mobile Hungria e British Telecommunications, entre muitos outros.

2.8 TIBCO PortalBuilder

A TIBCO (TIBCO, 2008a) é uma empresa especializada em produtos SOA e modelos de processos de negócio (*business process model* – BPM).

O PortalBuilder permite construir vistas personalizadas de informação integrada de diversos sistemas e processos, para os diferentes utilizadores do sistema.

Esta solução suporta o standard JSR-168, pelo que é capaz de consumir e publicar *portlets* WSRP. Suporta ainda SOAP, WSDL, *Universal Description Discovery and Integration* (UDDI), XML, J2EE, JDBC e *Representational State Transfer* (REST). Permite ainda a descoberta e apresentação de *web services* internos e externos.

A administração do sistema é feita usando uma interface web, permitindo delegar nos utilizadores capacidades de administração de secções do portal.

O sistema permite o uso de aplicações e *portlets* com interfaces interactivas (*rich interactive applications* – RIA).

O portal suporta múltiplos métodos de autenticação e permite o uso de *Single Sign On* (SSO).

Estão também disponíveis neste sistema ferramentas de gestão de conteúdos e ferramentas colaborativas.

Da sua carteira de clientes, na área de portal, destacam-se o Banco Central do Brasil, Cidade de Detroit, Reuters e Seagate.

2.9 Comparação de sistemas

Os sistemas apresentados têm características que os distinguem e características que os aproximam. A Tabela 1 procura representar de forma sucinta os elementos mais relevantes das diferentes arquitecturas, evidenciando semelhanças e diferenças entre os diferentes sistemas.

	Componentes gráficos	Serviços necessários	Suporte WSRP	Dados de fontes externas
Vignette	Java portlets	Servlet Application Server	Sim	ODBC, SQL, Web-services
BaSeWeP	Folhas de apresentação, gerando uma Java Server Page dinamicamente	Servlet Application Server (Tomcat, por exemplo)	Não	camada intermédia acessível via web-services, ODBC, SQL e Web-Services
uPortal	Java portlets	Servlet Application Server	Sim (2.4) Não (3.0)	Canais, ODBC, SQL, Web-services
SharePoint	Web parts	Internet Information Server	Sim	ODBC, SQL, Web-services
BEA	Java portlets	Servlet Application Server	Sim	ODBC, SQL, Web-services
TIBCO	Java portlets	Servlet Application Server	Sim	ODBC, SQL, Web-services

Tabela 1 - Comparação dos diferentes portais apresentados.

2.10 Conclusão

Como referido no início do capítulo, a personalização de sistemas web pode estar presente quer em portais, quer em CMSs. No entanto, neste capítulo, apenas são apresentados portais web. Isto porque no caso dos CMSs, a gestão de utilizadores é mais rígida e o grau de personalização permitido é mais limitado do que nos portais web. Uma vez que os CMSs não fornecem o nível de personalização pretendido, estes foram preteridos a favor dos portais web.

No caso dos portais web, apesar da lógica de funcionamento ser independente dos componentes, a lógica de apresentação tem que ser definida na própria interface do portal. Deste modo, se um determinado componente for configurado para ser disponibilizado apenas a um grupo de utilizadores, sempre que um qualquer utilizador aceder à página em questão, o servidor terá que verificar se este pertence ou não ao grupo que pode ver o componente.

O que se pretende no caso da Universidade é exactamente o contrário: o servidor deve interpretar o perfil do utilizador, que lhe deve indicar quais os componentes a carregar em

cada instante. Deste modo, toda a gestão de componentes é independente do portal. Será no perfil do utilizador que estará toda a configuração da interface que o portal deverá mostrar. No capítulo 5 – “O Portal my.ua”, será discutido o protótipo desenvolvido e que responde às necessidades aqui expressas.

3 Single Sign-On

O *Single Sign-On* (SSO), é um mecanismo que, num ambiente multi-sistema, possibilita que um utilizador introduza as suas credenciais de acesso apenas uma vez num dos sistemas de informação e, com isso, consiga a partir desse momento, acesso legitimado a todos os demais sistemas e computadores a que tenha permissão de acesso, sem necessitar de reintroduzir as suas credenciais. É um cenário cada vez mais utilizado pelas federações de identidades. Nestas, os diferentes sistemas de informação acordam entre si a possibilidade dos seus utilizadores acederem aos serviços disponibilizados pelos outros sistemas, usando para isso o seu login e palavra-chave do sistema de origem.

Além de reduzir potenciais quebras de segurança, este mecanismo diminui os erros de autenticação por parte dos utilizadores, e liberta a capacidade de processamento do sistema para outras tarefas que não a repetitiva autenticação de utilizadores. O SSO preocupa-se apenas com a parte de autenticação de utilizadores. A parte de validação ou autorização de acesso é da responsabilidade das aplicações cliente.

Neste capítulo serão abordadas algumas das alternativas mais relevantes no que concerne a SSO, quer do ponto de vista de normas, quer do ponto de vista de aplicações.

3.1 *Security Assertion Markup Language*

O *Security Assertion Markup Language* (SAML (Cover, 2008; Madsen & Maler, 2005)) é um standard, baseado em XML, para comunicação de autenticação, privilégios e características de utilizadores. Com base nesta troca de informação, um sistema A pode aferir informação acerca da identidade, características e direitos de um sujeito associado a um sistema B, em que A e B são independentes entre si.

O SAML é composto por quatro blocos: alegações, protocolos, ligações e perfis.

Por **alegação** entende-se um conjunto de informações fornecido por uma entidade a outra acerca de um dado sujeito, normalmente em cenários de pedido/resposta. As informações podem ser relacionadas com autenticação, atributos ou decisão de autorização.

Tipos de alegações:

- Autenticação – informação relacionada com a autenticação de um determinado sujeito, num determinado momento. Este tipo de alegação é produzido por uma autoridade SAML, denominada de fornecedor de identidades. É sua a responsabilidade de autenticar utilizadores e manter um registo de informações relacionadas com a sua sessão, enquanto a autenticação estiver válida.
- Atributo – informação que relaciona um determinado sujeito, com um determinado atributo ou característica. Em cenários em que a identidade do sujeito é irrelevante ou quando o anonimato deve ser preservado, é possível efectuar autorização de acesso baseada em atributos.
- Decisão de autorização – informação sobre o resultado do pedido de acesso a um determinado recurso por parte de um utilizador.

No bloco de **protocolo**, as mensagens protocoladas definidas no SAML permitem uma série de pares pedido/resposta para efectuar todas as operações previstas:

- pedir a uma entidade alegações sobre um sujeito,
- pedir uma autenticação,
- pedir um fecho de sessão de utilizador,
- etc..

O bloco de **ligações** define como se podem ligar as mensagens protocoladas a normas de comunicação entre sistemas. A título de exemplo, a ligação SAML SOAP define como as mensagens SAML podem ser transmitidas em envelopes SOAP.

Um **perfil** SAML define um conjunto de restrições e/ou extensões para se poder atingir um determinado fim. A título de exemplo, o perfil Web Browser SSO especifica como é que um fornecedor de identidades e uma aplicação web consumidora de identidades deverão interagir para fornecer um serviço de SSO ao utilizador.

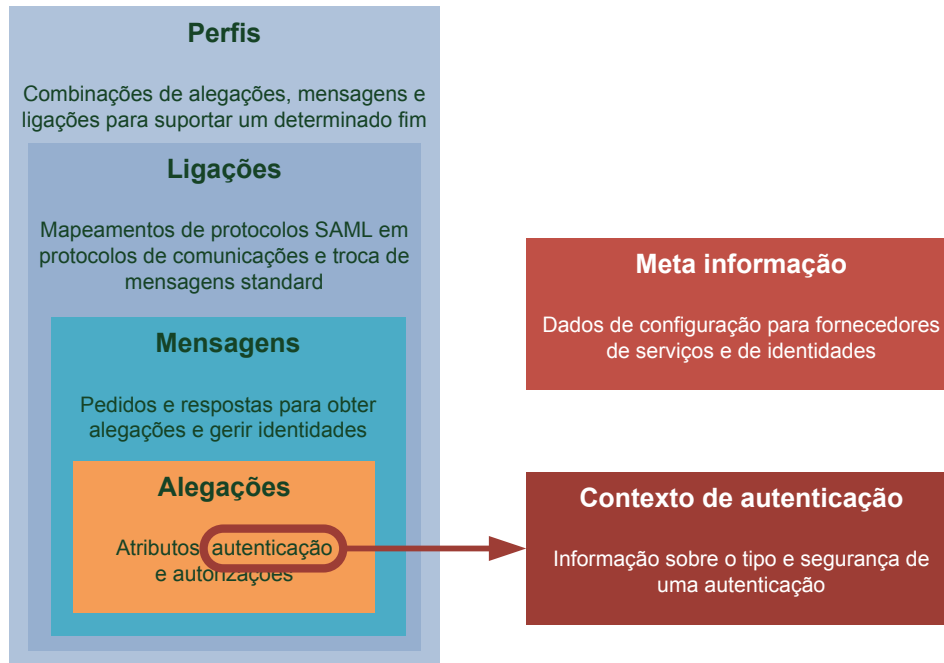


Figura 5 - Relação entre os diferentes blocos que definem o SAML

Além destes quatro blocos principais, existem mais dois blocos não essenciais (mas úteis) para um ambiente SAML: meta informação e contexto de autenticação.

O bloco de **meta informação** contém informação partilhável com outros agentes SAML sobre que atributos podem ser usados na comunicação com este agente, quais os métodos de encriptação suportados, quais as ligações suportadas, etc.

O bloco de **contexto de autenticação** contém informação respeitante ao tipo e segurança da autenticação de um determinado utilizador. Em certos cenários, um fornecedor de serviços pode ter necessidade de saber exactamente com que qualidade de credenciais um utilizador se autenticou, para que, caso a qualidade não seja aceitável, pedir que este se autentique utilizando um método mais seguro. Esta informação é enviada na alegação de autenticação.

A Figura 5 esquematiza a relação e dependência entre os diferentes blocos.

Com a versão 2.0, em 2005, surgiram novas características: pseudónimos, gestão de identidades, meta dados, encriptação, perfis de atributos, gestão de sessão, dispositivos móveis, mecanismos de privacidade e descoberta de fornecedores de identidades.

3.2 Normas de alto nível

O SAML é uma norma de baixo nível, flexível e extensível, pensado para ser usado por outras normas e aplicações: Liberty Alliance (Liberty Alliance Project, 2008), Web-Services Security (OASIS, 2006) e Shibboleth (Internet2, 2006). Estes, por sua vez, trouxeram melhorias para a versão SAML V2.0. Com base nas melhorias e na convergência de soluções, os diferentes sistemas tornaram-se compatíveis (pelo menos num determinado grau) entre si.

3.2.1 WS-Federation

O WS-Federation (Lockhart et al., 2006), uma iniciativa no âmbito de serviços Web, é uma especificação para federação de identidades, desenvolvida num consórcio de diversas empresas, entre as quais BEA, Microsoft, IBM e VeriSign. Esta especificação define os mecanismos que permitem que diferentes zonas de acesso condicionado interajam como federação, permitindo confiança de identidades, características e autenticação entre elas. Esta interacção baseia-se nos mecanismos definidos pela SAML, nomeadamente: WS-Trust (Nadalin, Goodner, Gudgin, Barbir, & Granqvist, 2007b), WS-Security (OASIS, 2006) e WS-SecurityPolicy (Nadalin, Goodner, Gudgin, Barbir, & Granqvist, 2007a). A Figura 6 esquematiza as dependências entre os diversos componentes necessários na WS-Federation.

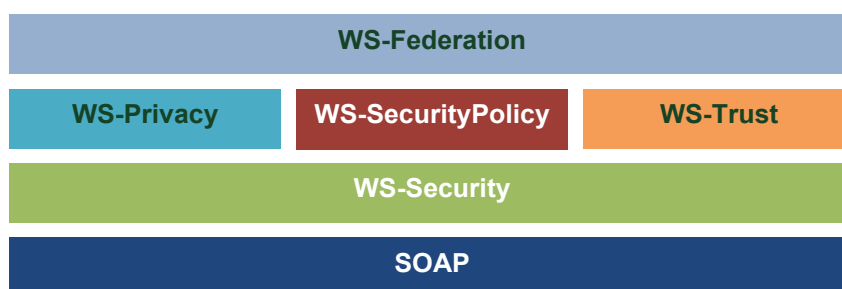


Figura 6 - Diagrama da componentes associados a WS-Federation

O **WS-Trust** especifica como se gere a noção de confiança entre partes. Com base nesta especificação, as partes envolvidas podem tomar como verdadeiras as afirmações proferidas pela contra-parte. Este processo baseia-se em chaves ou certificados válidos num espaço temporal finito, denominados de *tokens* ou testemunhos. Um testemunho é uma colecção de afirmações assinadas digitalmente. Estes certificados são gerados pelo *Security Token Service* (STS). Assim que o receptor da mensagem reconhece a validade do

certificado que a acompanha (e do seu emissor), vai tomar como fidedignas as afirmações e/ou indicações contidas na mensagem. A Figura 7 representa o modelo do STS (Goudner, Hondo, Nadalin, McIntosh, & Schimdt, 2007). Quem desejar aceder aos recursos disponibilizados pelo *Resource Provider* (RP), entra em contacto com o STS para obter o testemunho que poderá então apresentar ao RP. Por sua vez, este pode verificar a validade do testemunho recebido directamente com o STS.

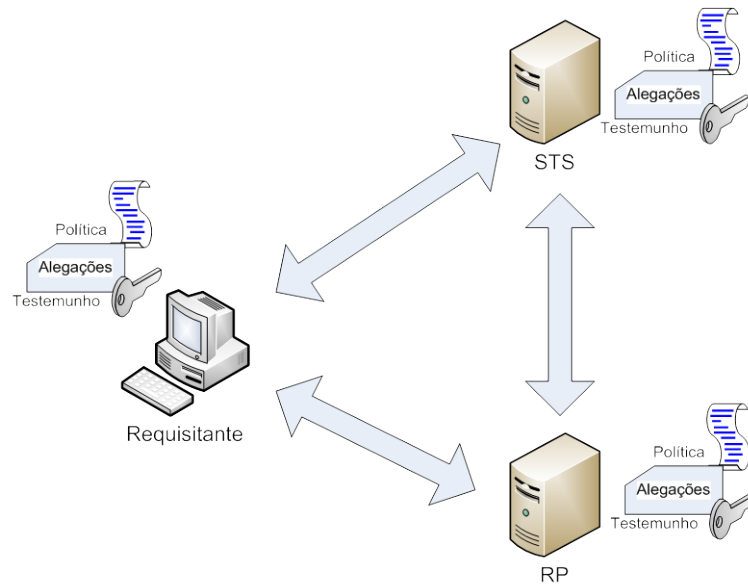


Figura 7 - Modelo do STS (Goudner et al., 2007)

O **WS-Security** especifica mecanismos para garantir autenticidade, integridade e confidencialidade das mensagens trocadas via SOAP entre as partes, recorrendo ao uso de testemunhos. Estes são enviados como alegações (e consequentemente como perfis) SAML. O WS-Security define uma plataforma para protecção de mensagens SOAP, em que os níveis de protecção são especificados em perfis determinados pela natureza do próprio testemunho. Estes perfis podem definir formatos como X.509 (Hallam-Baker, Kaler, Monzillo, & Nadalin, 2004; IETF - Internet Engineering Task Force, 2008) ou bilhetes Kerberos (Neuman & Ts'o, 1994).

O **WS-SecurityPolicy** permite a descrição de políticas de segurança para acesso aos serviços disponibilizados pelas partes. Mediante o uso de serviços web, e com base nestas descrições, as partes são capazes de reconhecer os tipos de testemunhos e afirmações necessárias para poderem trocar mensagens de uma forma segura.

Quando uma das partes (requisitante) pretende aceder/alterar informação existente noutra parte (fornecedor), começa por interrogar o fornecedor, sobre as políticas de acesso exigidas por este, para efectuar a operação pretendida. Usando WS-SecurityPolicy, o requisitante verifica se tem uma chave ou certificado válido para a operação em causa. Caso não tenha, deverá pedi-lo ao STS. Este pode ter sido mencionado na resposta dada pelo fornecedor. Caso o STS forneça a chave ao requisitante, este poderá então enviar a mensagem inicial ao fornecedor. O fornecedor, ao receber a chave e a mensagem, pode consultar o STS para validar a chave e as afirmações contidas na mesma, a fim de validar o emissor da mensagem e, finalmente, agir conforme pedido pelo requisitante.

Para que o testemunho gerado numa zona de informação seja válido noutra zona, é necessário que haja:

- uma relação de confiança entre os domínios,
- acordo acerca da sintaxe e semântica dos testemunhos e, finalmente,
- pontos de acesso para obter informações sobre requisitos de segurança exigidos e para pedir testemunhos.

Neste contexto, uma federação é um conjunto de zonas de informação distintas que acordaram na definição de políticas de troca segura de mensagens e pedidos entre si. A federação baseia-se na confiança, baseada na credibilidade, que se pode atribuir a afirmações vindas de uma zona de informação distinta.

A arquitectura subjacente a uma federação permite ao utilizador efectuar um *Single Sign Out*, ou seja, numa única operação, forçar todas as partes da federação a eliminar a informação sobre as suas credenciais de acesso que tenham armazenado. Além disso, os testemunhos gerados para este utilizador perdem automaticamente a sua validade.

Algumas das inovações desta especificação foram incorporados na SAML V2.0.

3.2.2 Liberty Alliance

A Liberty Alliance é um consórcio de indústrias que se uniram para a definição de standards para federação de identidades, SSO e acesso condicionado pelo utilizador a características intrínsecas a este. Entre os dinamizadores da iniciativa encontram-se Intel, Oracle e Sun. Os blocos funcionais da proposta da Liberty incluem a Identity Federation Framework (ID-FF) e a Identity Web Services Framework (ID-WSF). A primeira tem por

base a versão 1.x do SAML, usando para tal Perfis SAML. Sobre esta base, definiram camadas extra de funcionalidades e segurança. A segunda usa as alegações produzidas pelo SAML V2.0 como formato para os testemunhos de segurança, usados na comunicação de autenticações e autorizações, entre os seus actores.

Algumas das inovações produzidas na primeira versão por este consórcio, foram incorporadas no standard SAML V2.0, tais como, as adições ao SSO e o perfil de fim de sessão.

3.2.3 eXtensible Access Control Markup Language

A eXtensible Access Control Markup Language (XACML (OASIS, 2008)) é uma linguagem para controlo de acessos baseada em XML e definida pelo consórcio *Organization for the Advancement of Structured Information Standards* (OASIS). Esta destina-se a ser implementada em organizações que necessitem de um ponto central de configuração de políticas de segurança. Entre os principais impulsionadores do projecto estão empresas com IBM, SUN e BEA.

A ideia subjacente a esta norma é facilitar a gestão de organizações no que à segurança diz respeito. À medida que a dimensão de uma organização cresce, aumentam os problemas relacionados com segurança e permissões de acesso a informação. Desde os programas instalados nos computadores dos trabalhadores da organização, até aos documentos que lhes são enviados por correio electrónico ou feitos acessíveis na rede interna, todas as permissões de acesso e graus de confiança têm que ser configurados. No caso do software, estas definições são herdadas da plataforma onde os programas são executados. Assim, as políticas de segurança da plataforma terão que ser definidas ou herdadas de outro sistema. No caso de utilizadores, poderá ser necessário definir políticas de segurança em função do meio de acesso aos recursos: seja através do computador do escritório, rede sem fios, rede privada virtual, etc. Estas definições são, na maioria dos casos, configuradas de forma independente, o que torna alterações a políticas de segurança uma tarefa complexa e muitas vezes de resultados imprevisíveis. Do ponto de vista de gestão, é também complicado, se não impossível, saber qual a política geral de segurança efectiva para um determinado recurso ou imposta a um determinado utilizador.

Basicamente, o XACML é composto por três blocos principais (Regra, Política e Conjunto de Políticas) e por três elementos (ou pontos) principais (Ponto de administração de política – PAP –, Ponto de decisão de política – PDP – e Ponto de execução de política – PEP).

Os elementos ou pontos principais contêm interfaces que permitem administrar, decidir ou executar as políticas (em função do tipo de ponto).

A Figura 8 representa de forma simplificada o funcionamento do XACML, desde o pedido de acesso por parte de um utilizador, até ao momento em que efectivamente lhe acede.

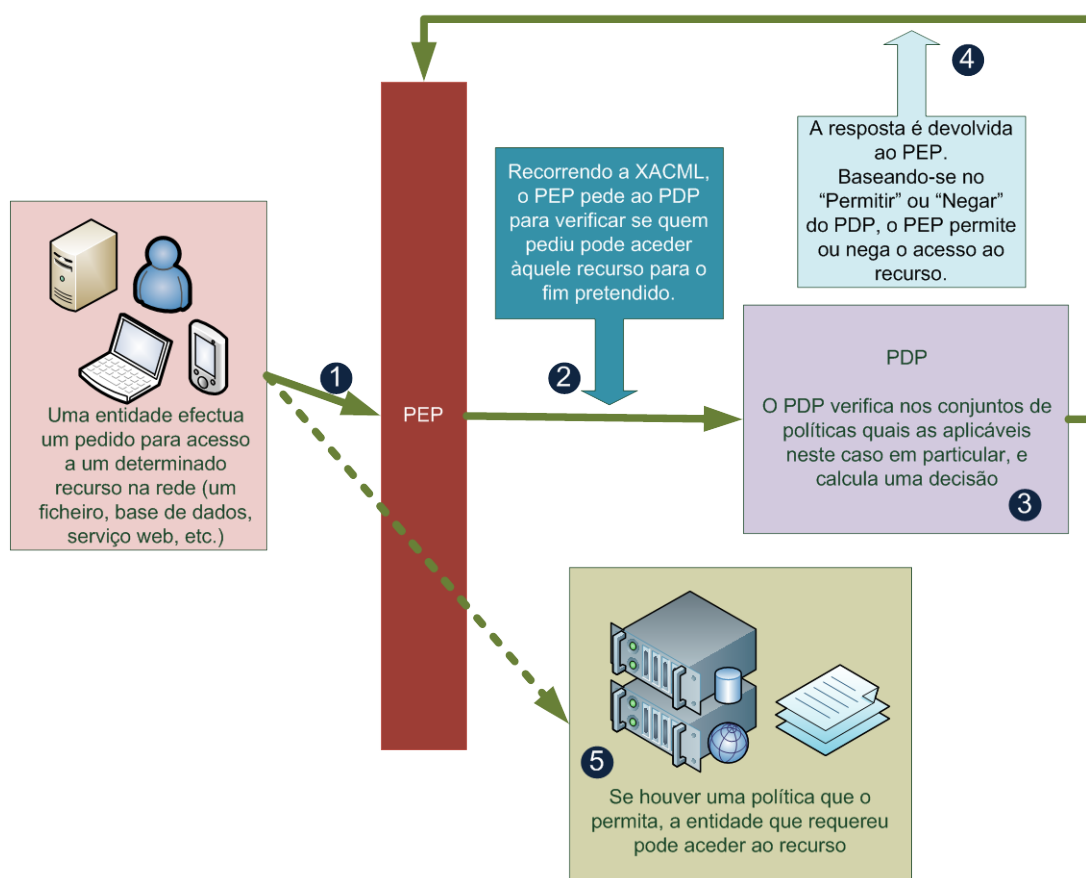


Figura 8 - Diagrama de funcionamento do XACML

A **regra** contém uma expressão booleana, que deverá ser apenas conhecida pelos PAP. As regras são definidas como condições sobre recursos, a que uma determinada entidade pretende aceder para um dado fim. A verificação de uma regra produz como consequência “permitir” ou “negar”.

A **política** contém um conjunto de regras e um algoritmo para a conjugação dos efeitos das diferentes regras aplicadas. A mesma regra pode usada em várias políticas.

O **conjunto de políticas** contém políticas e conjuntos de políticas aplicados segundo um algoritmo também especificado. A mesma política pode ser usada em vários conjuntos de políticas.

As versões recentes do XACML e do SAML foram pensadas para serem totalmente compatíveis.

3.3 Aplicações SSO

3.3.1 Shibboleth

Um *Shibboleth* é considerada uma palavra passe verbalizada, que permite distinguir indivíduos de grupos distintos. A história do Shibboleth remonta aos tempos bíblicos do Antigo Testamento. *Shibboleth* em dialecto Hebraico arcaico significava grão ou espiga. Na história, duas tribos inimigas travam uma batalha, da qual Guilead sai vencedora e monta um bloqueio para apanhar inimigos desertores.

“Guilead apossou-se dos vaus do Jordão, do lado de Efraim; sempre que um fugitivo de Efraim dizia: «Deixai-me passar», respondiam-lhe os homens de Guilead: «Tu és de Efraim!» Ele retorquia: «Não sou, não!» Então, diziam-lhe: «Pois então diz, por favor, ‘Chibolet’»; ele, porém, dizia: ‘Sibolet’, porque não era capaz de pronunciar correctamente. Então, prendiam-no e degolavam-no junto aos vaus do Jordão. Nesse dia, morreram quarenta e dois mil homens de Efraim.” (Jz 12, 5-6);

Passados os tempos de guerra, a expressão “shibboleth” passou a designar uma chave que identifica uma pessoa estranha a um grupo, ou por outro lado, que identifica indivíduos do mesmo grupo.

O sistema Shibboleth, desenvolvido pelo consórcio americano Internet2, é um programa de código livre baseado em normas e que fornece um SSO na Web em cenários intra e inter-organizações (Internet2, 2006). Foi inicialmente pensado para controlar o acesso a recursos disponibilizados na internet, de acordo com as necessidades das instituições de ensino superior, investigação e parceiros.

Este software usa activamente SAML na sua troca de mensagens. A partir da versão 1.3b tornou-se compatível com WS-Federation. Um dos pontos centrais deste software é a existência de uma gestão activa de privacidade na sua arquitectura. Além disso, permite administração federada e controlo de acesso baseado em características.

Os blocos principais deste software são: fornecedor de identidades (*identity provider – IP*), fornecedor de serviços (*service provider – SP*) e ainda um “de onde é” (*where are you from – WAYF*).

O *Identity Provider (IP)* gere e mantém credenciais e atributos de utilizadores. Quando lhe é pedido, o IP verifica a validade das credenciais do utilizador ou entrega atributos de utilizadores aos SP. Um IP contém uma autoridade de autenticação, responsável por validar e enviar alegações de autenticação. Contém também um serviço de SSO e ainda uma autoridade de atributos, responsável por validar e enviar alegações de atributos.

O *Service Provider (SP)* gere recursos que devem ser acessíveis apenas a entidades autorizadas.

O WAYF é um elemento opcional, servindo de proxy entre o SP e IP. O SP pode enviar o cliente para um WAYF, onde este escolhe qual o IP onde deseja efectuar a autenticação. Após a escolha, será então reencaminhado para o IP seleccionado.

A Figura 9 esquematiza o funcionamento básico deste serviço, imediatamente após a tentativa de acesso a um recurso protegido do SP, por parte de um utilizador.

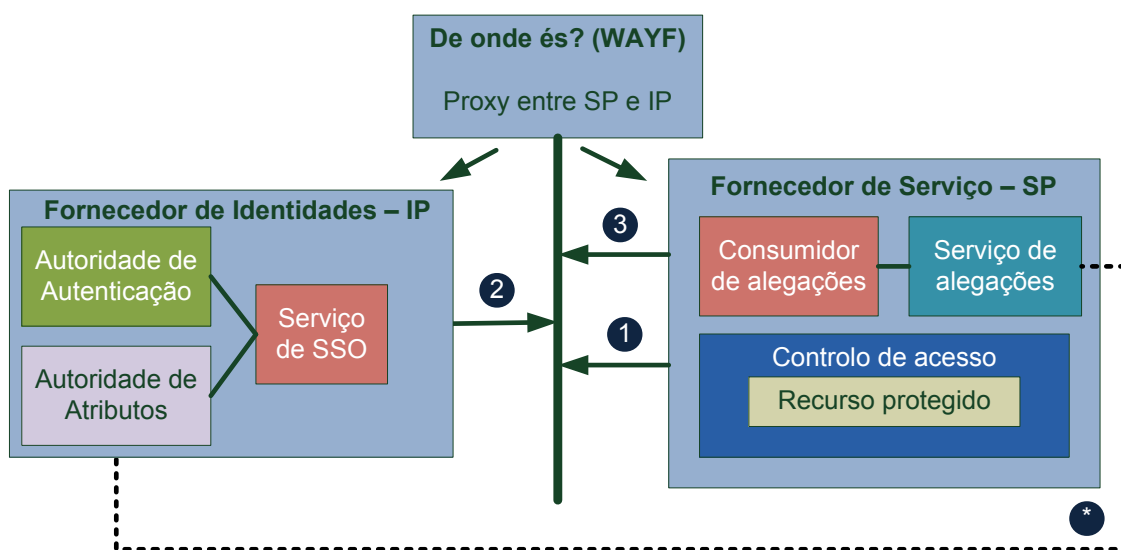


Figura 9 - Esquema de funcionamento do Shibboleth

Legenda da Figura 9:

1. Redirecionamento via URL para o SSO (ou WAYF). Possível passagem de parâmetros identificativos do SP, a localização do consumidor de alegações e opcionalmente, a URL de destino para o cliente, após correcta validação.
 2. Após validação, o IP constrói uma alegação de autenticação SAML que é enviada para o consumidor de alegações com indicação da chave temporária do utilizador, via HTTP Post do navegador de internet do cliente.
 3. Após recepção do SAML, o SP pode verificar mais alegações sobre o cliente, para poder decidir, ou delegar a decisão de acesso ao recurso protegido.
- *. Após validação, os pedidos de alegações posteriores podem ser efectuados por um canal exterior, com comunicação directa entre SP e IP.

Os elementos SAML são usados de forma intensiva nesta aplicação. Uns na sua forma original, outros aumentando ou modificando as capacidades do standard em causa. Alegações SAML de autenticação (aquando da autenticação do utilizador), alegações SAML de atributos (aquando da verificação de alegações sobre o utilizador) e alegações SAML de autorizações, todas são usadas pelo Shibboleth.

3.3.2 CardSpace

O CardSpace (Chappell, 2008) é um sistema compatível com a WS-Security. Trata-se de um produto da Microsoft, cujo funcionamento é em tudo semelhante ao STS do modelo WS-Security. O utilizador pode ir armazenando no seu CardSpace as suas identidades digitais (incluindo atributos), sejam elas certificados digitais, pares utilizador/chave, Como atributos, cada uma das identidades pode armazenar dados distintos, desde primeiro e último nome, até número de cartão de crédito, que poderão ser fornecidos ao serviço que pediu a identidade. Cabe ao serviço confiar na informação, após a validação dos dados de autenticação.

Quando um utilizador pretende, por intermédio de uma qualquer aplicação que necessite de o identificar (seja um sítio na internet ou uma aplicação local), aceder a informação de um fornecedor de dados externo, os passos a seguir são os exemplificados na Figura 10.

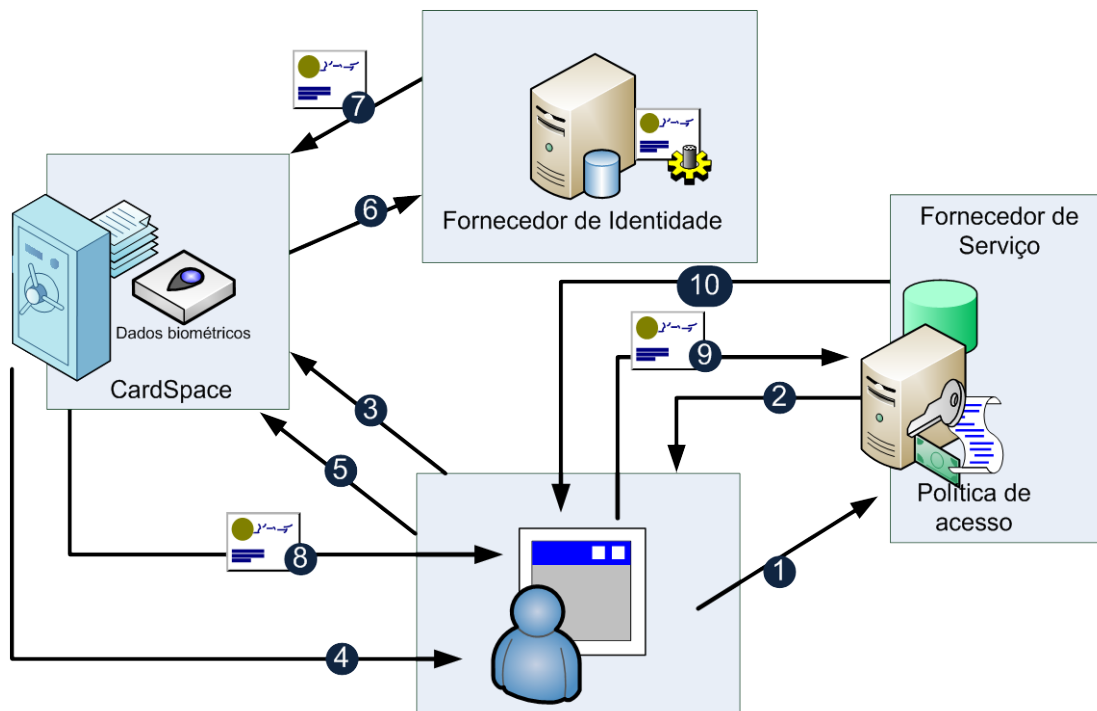


Figura 10 - Fluxo de troca de mensagens para acesso a recursos protegidos, usando o CardSpace

1. O processo é desencadeado quando um utilizador, por intermédio de uma aplicação (web ou instalada no seu computador) pretende aceder a um recurso de fornecedor externo.
2. O fornecedor responde indicando que o recurso é protegido, sendo necessário obedecer aos critérios definidos na política de acesso para a operação em causa. Esta informação está presente na política de acesso do fornecedor e inclui elementos como formatos aceitáveis para os testemunhos de segurança (simples nome de utilizador, certificados X.509, bilhetes Kerberos, chaves SAML ou outro qualquer formato) e que afirmações ou características do utilizador deverão acompanhar o testemunho.
3. Ao receber esta informação, a aplicação pede ao CardSpace um testemunho que tenha as características pedidas.
4. O CardSpace pode pedir ao utilizador para seleccionar uma identidade digital (dentro das que possuem as características pedidas).
5. O utilizador selecciona uma identidade.
6. Com base na identidade seleccionada, o CardSpace pede, ao fornecedor da identidade, o testemunho exigido pelo fornecedor de dados.

7. O fornecedor de identidade entrega ao CardSpace o testemunho em causa.
8. O CardSpace encaminha o testemunho recebido para a aplicação que o pediu.
9. A aplicação, por sua vez, entrega o testemunho ao fornecedor de dados para poder continuar com a operação que originou o pedido.
10. O fornecedor de dados verifica o testemunho e, confiando nos predicados associados a este, permite continuar a operação.

3.3.3 Windows Live ID

Trata-se de um sistema centralizado de autenticação de utilizadores, criado e mantido pela Microsoft (Microsoft Corporation, 2006). O sistema permite aos utilizadores aceder a diversos sites, com um único par nome de utilizador/palavra-chave. O Windows Live ID evoluiu do Microsoft Passport, sendo esta a proposta da Microsoft para um fornecedor de identidades, atribuindo ao utilizador identificadores únicos por cada par utilizador/aplicação. Desta forma, as aplicações associadas a este serviço não necessitam de gerir contas de utilizador ou chaves de acesso.

O Live ID permite às aplicações o acesso a dados pessoais dos utilizadores, através de um processo denominado de delegação de autenticação. Neste processo, o sistema permite o acesso a dados da conta do utilizador em causa, mas sempre de forma condicionada à prévia aprovação por parte do utilizador. A única informação disponibilizada de forma automática é o identificador, sinal de que as credenciais usadas são válidas (mesmo o email/login usado não é disponibilizado). Pode ser usado para, por exemplo, dar acesso aos contactos de *MSN Messenger* de um utilizador, para construção de uma rede social numa aplicação.

A delegação de autenticação funciona numa perspectiva temporal: o utilizador dá o seu consentimento à aplicação para aceder aos dados em questão, durante um determinado período temporal. Este consentimento pode ser revogado pelo utilizador, a qualquer momento, nos serviços Web do Live ID.

O fluxo de mensagens que permite a uma aplicação, o acesso a dados de um utilizador, para um determinado fim, está esquematizado na Figura 11.

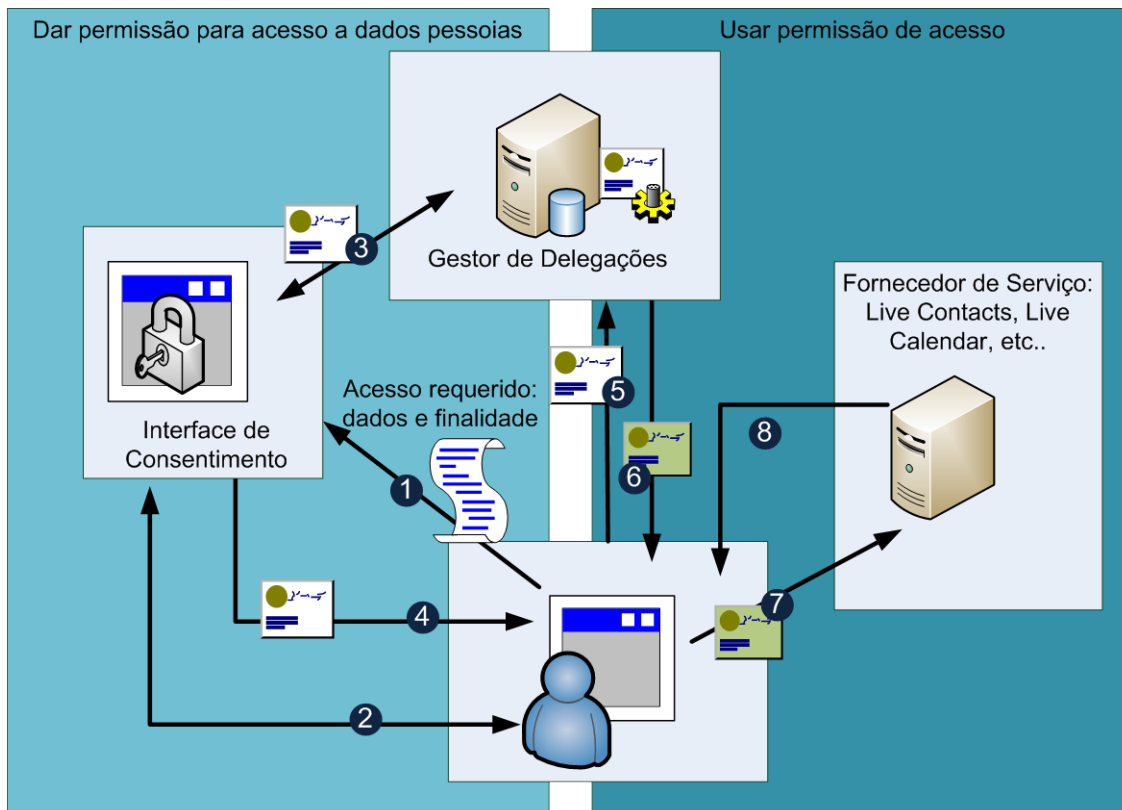


Figura 11 - Fluxo de mensagens para permitir o acesso de uma aplicação a dados do utilizador

A Figura 11 é constituída por dois momentos distintos: os passos 1 a 4 e os passos 5 a 8.

Os passos 1 a 4 representam a fase em que o utilizador, por ter de dar o seu consentimento, tem que estar ligado na aplicação que necessita de aceder aos seus dados pessoais. Os passos são explicados em seguida:

1. A aplicação indica à interface de consentimento a que dados pretende aceder e para que fim.
2. A interface pede ao utilizador para indicar se dá ou não o seu consentimento, e se der, durante quanto tempo.
3. A interface pede ao gestor de delegações um testemunho de delegação
4. O testemunho é entregue à aplicação, que o guarda para utilizações futuras

Assim que a aplicação tem acesso ao testemunho, pode imediatamente passar para o passo 7. No entanto, o testemunho de delegação recebido tem uma validade temporal finita (inferior à validade definida pelo utilizador), pelo que num cenário normal, antes de efectivamente aceder aos dados, os passos são:

5. A aplicação pede ao gestor de delegações um testemunho actualizado, enviando como prova que o pode fazer, o testemunho que lhe foi entregue anteriormente (passo 4 ou 6 numa iteração anterior).
6. O gestor entrega um testemunho actualizado à aplicação
7. A aplicação envia o pedido acerca dos dados do utilizador ao fornecedor de serviços em causa, com o testemunho anexado
8. Finalmente, o fornecedor de serviços executa o pedido da aplicação.

O Live ID pode ser acedido usando WS-Trust, CardSpace ou mesmo *Remote Authentication Dial In User Service* (RADIUS (Rigney et al., 2000)).

O Windows Live ID pode ser usado num cenário de federação de identidades, recorrendo para tal a WS-Federation.

3.3.4 Eduroam

Eduroam (EduRoam, 2008) é um acrónimo para designar mobilidade em educação e pesquisa (EDUcation ROAMing). Trata-se de uma parceria que permite a autenticação de utilizadores de uma instituição de ensino, enquanto estes estão noutra instituição, garantindo-lhes assim privilégios de acesso a serviços de rede e sistemas.

Usando servidores RADIUS, as credenciais do utilizador são validadas na sua instituição de origem e o resultado desta, juntamente com mais alguns atributos ou características do utilizador, são encaminhados para a instituição de acolhimento, a fim de permitir ou não o acesso ao serviço em questão.

O Eduroam é organizado em confederações. Em 2008 existem duas confederações: a eduroam Europa e a eduroam Ásia-Pacífico. A Figura 12 esquematiza o modelo estrutural do Eduroam, com indicação das dependências entre as diversas partes.

Os servidores RADIUS ao nível da confederação, são responsáveis pelo encaminhamento dos pedidos entre os servidores de nível federativo ou nacional. Por sua vez, os servidores de nível federativo, reencaminham os pedidos dentro da sua federação (tipicamente no seu país).

Finalmente, cada instituição tem de ter um servidor RADIUS, um fornecedor de identidades e um fornecedor de serviços. Cabe ao fornecedor de identidades da instituição de origem, validar as credenciais de acesso introduzidas pelos seus utilizadores (seja na

própria instituição ou em instituições visitadas). Com base na resposta dada pelo fornecedor de identidades, o fornecedor de serviços decide ou delega a decisão sobre a permissão de acesso ao recurso em causa, para serviços internos da instituição.

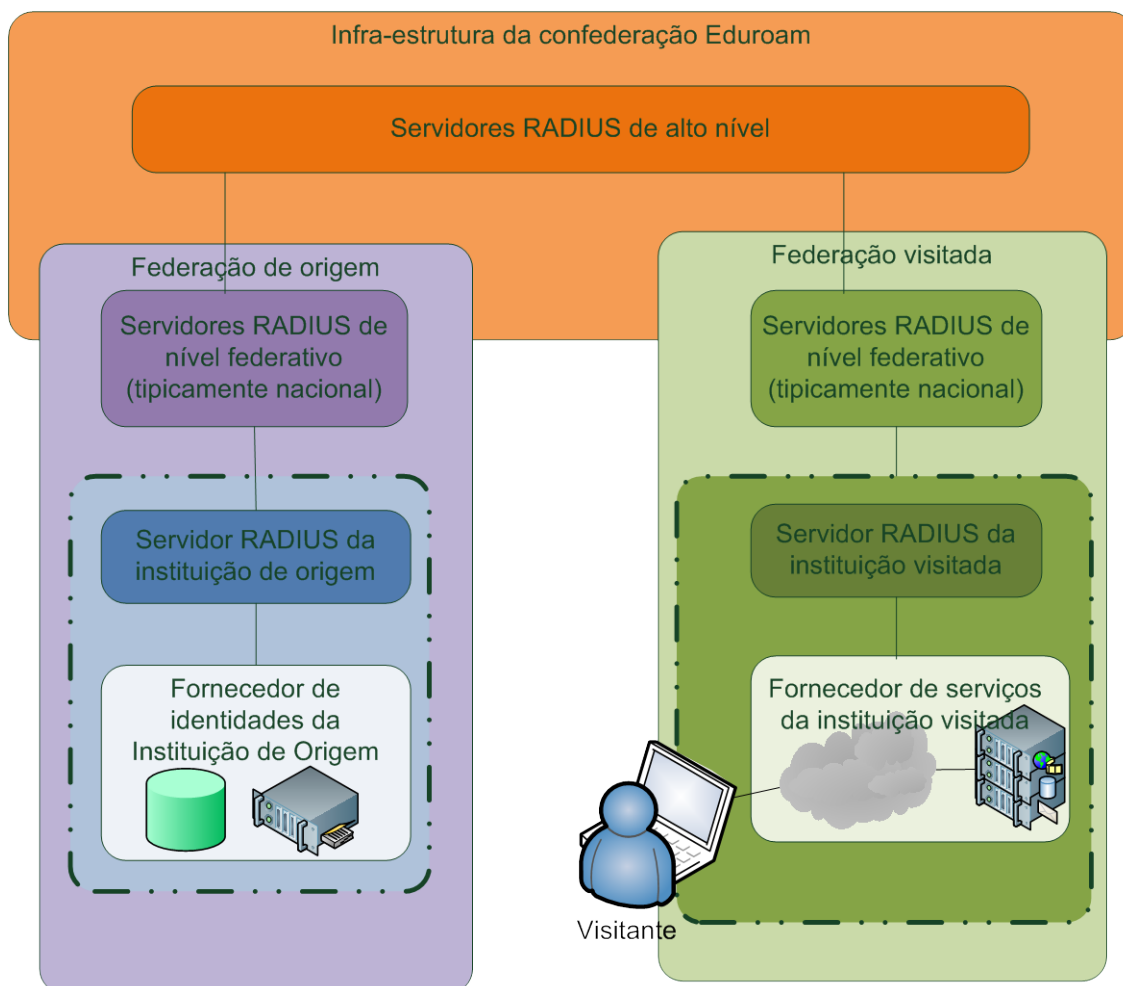


Figura 12 - Estrutura geral do Eduroam

Ao associar-se a uma confederação, as instituições concordam em confiar nas respostas que lhe são enviadas pelas restantes instituições.

4 Perfil de utilizador

“A personalização permite aos utilizadores obter informação adaptada às suas necessidades, objectivos, conhecimento, interesse e outras características” (Zimmermann, Specht, & Lorenz, 2005). Após identificação e validação do utilizador, é necessário que o sistema o descreva, de acordo com as características da aplicação a aceder. Esta operação é realizada recorrendo a um perfil de utilizador.

4.1 Noção de perfil de utilizador

Um perfil de utilizador é um modelo individual de utilizador, uma colecção de informação que descreve as necessidades, preferências e interesses de um utilizador em particular. É, no fundo, uma colecção de dados que descrevem um indivíduo de forma adequada a um sistema (G. Adomavicius, 1999). O processo de aquisição desta informação é chamado “*user profiling*”, sendo normalmente “baseado em conhecimento ou em comportamento” (Daigle & Cuocco, 2002; Middleton, Shadbolt, & Roure, 2004; Rich, 1979). A maioria dos sistemas usa o “*user profiling*” para tentar entender os interesses do utilizador e para lhe proporcionar a versão comercial da “informação certa para a audiência certa”, ou seja, publicidade.

No caso particular desta tese, a informação baseada em conhecimento, reporta-se a quem o utilizador é e a que serviços pode aceder na organização. A aquisição de informação baseada em comportamento não será considerada, ainda que possa ser interessante em cenários de personalização automatizados, como áreas de visita frequente ou uso de funções avançadas na aplicação. Esta rejeição prende-se com o facto do estudo apresentado ser sobre modelos de utilizador, numa perspectiva da organização. Logo as ferramentas, operações e informação que a organização fornece ao utilizador, não dependem do que o utilizador faz, mas de quem o utilizador é e ao que o utilizador pode aceder. Sobre *user profiling* comportamental, em (Peter, 2001) o autor explica hipermédia adaptado como uma alternativa a aplicações Web tradicionais, onde todos os utilizadores obtêm o mesmo tipo de informação.

Um modelo de utilizador pode ser classificado de acordo com três dimensões (Rich, 1979):

- utilizador geral ou modelo individual de utilizador;
- construído explicitamente pelo utilizador ou baseado no comportamento do utilizador;
- com informação altamente especializada e de curta duração ou com informação mais geral de maior duração.

Cada uma destas dimensões tem as suas falhas e benefícios:

- Confiar em informação introduzida pelo utilizador pode dar origem a informações erradas (seja por o utilizador não saber dar a resposta mais correcta no seu caso, ou simplesmente porque não deseja dar informações correctas). Construir um modelo baseado no comportamento pode também conduzir a fracos resultados.
- Um modelo canónico de utilizador é uma boa abordagem para generalização mas pobre em informação pessoal (o mesmo modelo será atribuído a todos os utilizadores). No caso do modelo individual de utilizador a situação é a inversa.
- Modelos com informação a curto prazo, altamente especializada, são importantes para determinadas aplicações mas podem correr o risco de negligenciar o histórico de informação do utilizador que poderia ser usado no processo de adaptação da aplicação.

Os dados de um perfil podem ser divididos em duas categorias (G. Adomavicius, 1999): demográficas e transaccionais. A primeira descreve quem o utilizador é e a segunda o que ele faz. Em sistemas de recomendação, por exemplo, a parte transaccional é o histórico de compras do utilizador.

4.2 Estereótipos

Um método rápido para construir um modelo de utilizador é usar a informação disponível (pessoal e profissional, interesses, demográfica, social ...) e fazer corresponder essa informação a um conjunto de características pré-determinadas. A estes conjuntos de características dá-se o nome de estereótipos (Kobsa, 1995; Rich, 1979). Às condições que definem como, e em que circunstâncias, um estereótipo é atribuído a um utilizador, dá-se o nome de condições de mudança (*trigger conditions* (Rich, 1979)). Um estereótipo define o

comportamento e semelhanças expectáveis entre utilizadores de uma mesma classe, e pode ser usado como um perfil de utilizador inicial.

O Grundy (Rich, 1979) foi uma aplicação desenvolvida para fazer o papel de bibliotecário, efectuando sugestões de leitura aos utilizadores. As sugestões são determinadas mediante a resposta a uma série de questões colocadas ao utilizador de forma interactiva. Cada questão colocada é fruto da análise das questões e respostas anteriores, pelo que à medida que o utilizador vai respondendo, o sistema vai activando e desactivando estereótipos (primeira sessão do utilizador ou utilizador familiarizado, até estereótipos mais particulares, relacionados com características pessoais introduzidas pelo utilizador – personalidade, religião, etc). Assim que o sistema consegue um conjunto suficiente de informação sobre o utilizador e consegue mapear essa informação nos estereótipos que possui, este passa a ter uma ideia mais concreta dos possíveis gostos do utilizador. Com base nesta crença, o Grundy sugere ao utilizador um livro para ler. Caso o utilizador não goste do livro escolhido, uma série extra de questões feitas ao utilizador e relacionadas com a recusa do livro, permite refinar alguns parâmetros das características associadas ao utilizador e sugerir-lhe um livro que seja mais do seu agrado.

<i>FACET</i>	<i>VALUE</i>	<i>RATING</i>
Activated-by	<i>Athletic-w-trig</i>	
Genl	<i>ANY-PERSON</i>	
Motivations		
Excite	3	600
Interests		
Sports	4	800
Thrill	5	700
Tolerate-violence	4	600
Romance	-5	500
Education	-2	500
Tolerate-suffering	4	600
Strengths		
Physical-strength	4	900
Perseverance	3	600
SPORTS-PERSON		

Tabela 2 - Exemplo de um estereótipo do Grundy

Neste sistema, um estereótipo é um conjunto de facetas e condições de atribuição, em que cada faceta é um tuplo[característica, valor, confiança]. A Tabela 2 apresenta um exemplo de um estereótipo do Grundy.

Quer o valor, quer a confiança, são representados em termos de escala (-5 a 5 e 0 a 1000, respectivamente). O montante atribuído a “valor” representa a afinidade do estereótipo com aquela característica. A confiança representa o grau de certeza que deve ser atribuído ao tuplo.

Neste sistema, todos os estereótipos derivam de um estereótipo base, denominado por “*ANY-PERSON*”. Este tem um conjunto de facetas genéricas, com valores de confiança suficientemente baixos, para serem facilmente substituídos por outros existentes em estereótipos mais específicos.

4.3 Sistemas de Modelação de Utilizador

Como referido por (Judy, Bob, & Piers, 2002; Kobsa, 2001; Kobsa & Fink, 2006), a maioria dos sistemas de modelos de utilizador são sistemas “concha”, ou seja, são sistemas adaptados para uso exclusivo de uma única aplicação, tornando-os ineficientes numa perspectiva de troca de informação e impraticáveis em cenários de várias aplicações. Os sistemas concha foram criados como peças reutilizáveis, para gestão de modelos de utilizador, por parte de uma aplicação em particular. Operações como representação do modelo de utilizador, manutenção, consistência de informação e classificação automática de utilizadores, são algumas das características destes sistemas.

No fundo, são sistemas funcionais, mas “vazios de informação”, até que sejam configurados pela aplicação, com informação específica sobre os modelos de utilizador: que grupos estão definidos, que características tem cada grupo, etc.

Há cada vez mais excepções, como mencionado em (Judy et al., 2002). Em (Kobsa & Fink, 2006) os autores apresentam o *User Modeling Server* (UMS), um servidor de modelos de utilizador baseado em LDAP. O UMS é uma extensão do esquema tradicional de LDAP para armazenar informações do utilizador.

4.3.1 General User Model System

O *General User Model System* (GUMS (Finin, 1989)) foi uma das primeiras tentativas de separar a modelação de utilizadores da lógica da aplicação. Apesar de pouco usado por aplicações, este sistema é considerado como o embrião dos servidores de modelos de utilizadores.

Cabe às aplicações a interacção com o utilizador, aquisição e processamento de dados e posterior entrega ao GUMS para actualização do modelo de utilizador. Sempre que necessário, o GUMS entrega informação sobre o utilizador às aplicações.

É da responsabilidade do GUMS a manutenção e consistência do modelo do utilizador. Se, por força dos dados que uma aplicação envia ao GUMS, este tiver sobre um determinado utilizador uma crença por omissão e se a aplicação envia novos dados sobre a mesma crença, mas com uma maior convicção, então, a actualização do modelo do utilizador é simples, visto que passa a valer a nova convicção. Mas, se pelo contrário, os dados enviados pela aplicação são contrários a uma crença inferida por meio de regras e estereótipos, então, será muito provavelmente necessário reclassificar o utilizador no universo dos estereótipos existentes, fazendo para isso o caminho inverso ao da dedução que conduziu à classificação errada. Além disso, o GUMS pode também inferir novos factos sobre o utilizador, baseando-se no conjunto de informações que dispõe sobre este.

Apesar de poder interagir com várias aplicações, o GUMS armazena um modelo de utilizador independente para cada uma delas.

Como forma de representação do modelo de utilizador, o GUMS usa três métodos distintos: estereótipos, regras explícitas por omissão e falha como negação. Os estereótipos fornecem as classes de utilizadores mais gerais, nas quais se podem definir regras explícitas que poderão fornecer uma melhor visão do modelo de um determinado utilizador. A falha como negação é, normalmente, usada como último recurso, permitindo saber que um determinado utilizador não faz parte de um estereótipo ou não obedece a uma determinada regra.

As responsabilidades do GUMS são (Tim & David, 1986):

- Manter uma base de informação dos factos observados pelo utilizador,
- Inferir factos adicionais baseando-se nos factos observados,
- Informar quando é que um determinado facto pode ser inferido ou assumido como verdadeiro pela aplicação,
- Manter a consistência da informação do modelo do utilizador e
- Reconhecer quando os factos observados, sobre um utilizador, são inconsistentes com o estereótipo em que este está inserido e fornecer alternativas de inclusão noutros estereótipos.

Internamente, o GUMS é um motor lógico, escrito em Prolog (Covington, Nute, & Vellino, 1996), que assume, para cada predicado, um de quatro valores possíveis: Verdadeiro, Falso (valores definitivos), Assumir como verdadeiro e Assumir como falso (valores por omissão).

Os estereótipos e regras são construídos através de predicados Prolog.

4.3.2 User Modeling Tool

O *User Modeling Tool* (UMT (Brajnik & Tasso, 1994)) é um sistema concha que permite a definição de estereótipos hierarquicamente ordenados, regras de inferência e de detecção de contradições entre o perfil do utilizador e o que é definido nos estereótipos. As premissas de informação que chegam a este sistema podem ser classificadas como definitivas (verdades absolutas sobre o utilizador) ou presumidas (premissas verdadeiras até ao momento, que podem vir a revelar-se como falsas e que podem ser removidas do conjunto de informações adquiridas do utilizador).

À semelhança de outros sistemas, os estereótipos permitem a definição de condições de atribuição: regras que definem os critérios para que um utilizador possa ser considerado como integrante do estereótipo. Quando a informação sobre um utilizador, em conjunto com o modelo de utilizador a que este se encontra sujeito na altura, se encontra dentro dos critérios de atribuição de um dado estereótipo, as premissas contidas no estereótipo são integradas no conjunto de informação acerca do utilizador. O UMT procede então à verificação da consistência de informação (os dados integrados podem ser contraditórios com a informação recolhida até ao momento sobre o utilizador) usando para isso as regras de inferência e de contradição definidas anteriormente. Com base nestas informações, o

modelo de utilizador usado (que define no fundo, quais as características do utilizador) poderá ser revisto, de entre os modelos existentes que se enquadrem com as informações adquiridas sobre o utilizador.

4.3.3 TAGUS

O Tagus (Paiva & Self, 1994) é um sistema concha para uso em cenários de elevada interactividade entre utilizador e aplicação, como é o caso de ambientes de ensino assistido por computador. Como parte do modelo de utilizador integra, também, informação sobre o desempenho do utilizador/aluno nas tarefas/aulas anteriores. À semelhança do UMT, permite a definição de hierarquias de estereótipos. Da sua arquitectura, faz parte um mecanismo de inferências, um sistema de manutenção de verdade (coerência da informação) e um sistema de diagnóstico. No sistema de manutenção de verdade, são permitidos vários níveis de confiança acerca das premissas sobre o utilizador. No sistema de diagnóstico, é permitido aos programadores da aplicação (mas não aos utilizadores) o escrutínio de dados dos utilizadores.

Implementado em Prolog e Gödel, armazena cada modelo de utilizador num ficheiro distinto que pode ser carregado de forma dinâmica, durante o ciclo de execução do Tagus. Uma das inovações deste sistema é a sua capacidade de fornecer à aplicação um perfil de utilizador simulado.

4.3.4 Personis

O sistema Personis (Judy et al., 2002) é uma das primeiras abordagens a um servidor de modelos de utilizador, passível de ser utilizado por diversas aplicações em simultâneo. Além disso, e à semelhança de alguns sistemas concha, permite ao utilizador verificar toda a informação que existe no sistema sobre o utilizador, incluindo as inferências feitas para associação do utilizador a grupos. Um ponto inovador neste sistema prende-se com a possibilidade do utilizador controlar a informação que, sendo sua, pode ser partilhada entre aplicações que usem o seu perfil do utilizador.

A Figura 13 apresenta um cenário de múltiplas aplicações a consultar o servidor de modelos e perfis de utilizador, em conjunto com a arquitectura do servidor. A cada aplicação é entregue uma vista do perfil global do utilizador, de acordo com as necessidades específicas da aplicação.

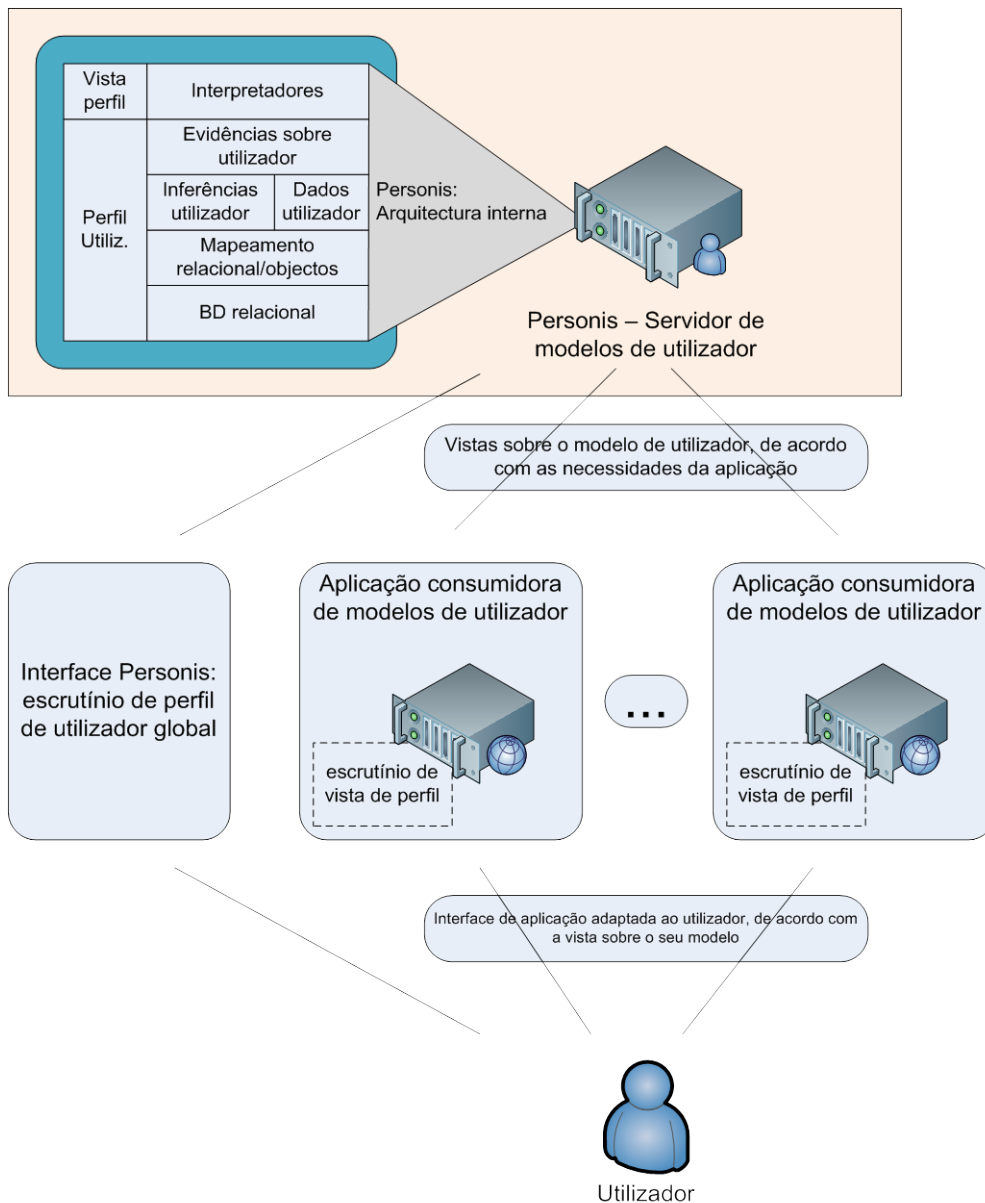


Figura 13 - Esquema de funcionamento do Personis em cenários multi-aplicação e arquitectura interna do servidor

As inferências são informações deduzidas pelas aplicações, durante a utilização das mesmas, por parte do utilizador. Os dados recolhidos são informações obtidas pelas aplicações sobre o utilizador: podem ser originários de formulários ou podem ser resultado da utilização da aplicação por parte deste. Cada evidência contém, além da informação própria da mesma, dados sobre a fonte que a originou.

Esta informação é armazenada numa base de dados relacional, sendo o acesso à mesma feita por uma camada intermédia de mapeamento entre os objectos (evidências) e a base de dados. Esta camada permite mapear parte do perfil do utilizador para servidores distintos,

promovendo cenários de perfis distribuídos. Cada associação entre evidência e aplicação, tem associadas regras de acesso para leitura e para escrita.

Uma vista do perfil é composta por um conjunto de componentes da arquitectura: interpretadores, evidências (registos) sob a forma de inferências, e/ou dados recolhidos sobre o utilizador. Um interpretador é um algoritmo específico que irá dar sentido às inferências produzidas pelas aplicações no perfil do utilizador. Sem estes, o perfil do utilizador não é mais que um conjunto de dados sobre o utilizador. A título de exemplo, pode-se considerar que um interpretador A, toma como garantidas inferências produzidas por uma aplicação X, enquanto um interpretador B, analisa as mesmas inferências com um grau de certeza muito menor. O resultado das análises ao perfil irá trazer resultados distintos, pelo que serão duas vistas diferentes sobre o mesmo perfil. Uma vista pode ser resultado da análise de vários interpretadores.

4.3.5 User Modeling Server

O *User Modeling Server* – UMS – (Kobsa & Fink, 2006) propõe o uso de directórios LDAP como servidores de modelos de utilizador. O pressuposto principal neste sistema prende-se com as potencialidades associadas aos directórios: escalabilidade, disponibilidade, extensibilidade e desempenho. Além disso, é tarefa dos directórios disponibilizar (e manter) informações sobre indivíduos, grupos e equipamentos, pelo que parte dos conceitos inerentes à modelação de utilizadores estão já presentes. A Figura 14 representa a arquitectura geral do UMS.

A arquitectura do sistema compreende um bloco central de componente de directório, e vários módulos que podem ser ligados ao servidor (componentes de modelação de utilizador), dedicados em exclusivo a uma determinada tarefa. A comunicação entre estes e o componente de directório, pode ser efectuada usando *Common Object Request Broker Architecture* (CORBA) com o módulo de planificação ou LDAP (com o módulo de comunicação).

A componente de directório é constituída pelos módulos de comunicação, representação e planificação (agenda).

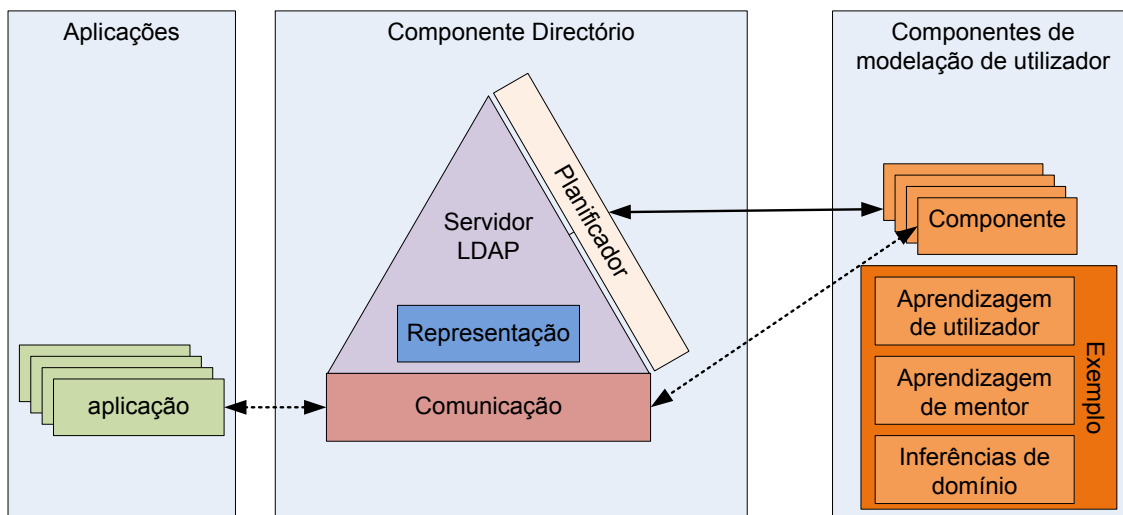


Figura 14 - Arquitectura geral do UMS

O módulo de comunicação é responsável pela interacção com as aplicações que necessitem de aceder a modelos de utilizadores e pela interacção entre os componentes de modelação de utilizador e o directório.

O módulo de representação é responsável pela gestão do directório. Cabe a este módulo gerir os diversos modelos de utilizador em uso no UMS. Uma vez que o UMS é baseado em LDAP, os modelos (ou perfis) de utilizador e perfis de uso são definidos de raiz, usando a classe de objectos padrão do LDAP. Os perfis de uso, funcionam como bases de armazenamento persistente de dados de utilização, fornecidas pelas aplicações ao UMS e como informação processada pelos componentes de modelação de utilizador.

O módulo de planificação encapsula o servidor LDAP usado como repositório do directório e orchestra a comunicação entre os diferentes módulos e componentes que fazem parte do UMS. Este módulo contempla também um notificador de eventos, visto que os componentes de modelação de utilizador podem subscrever determinadas mudanças no modelo dos utilizadores. Assim, a notificação de alterações é efectuada a intervalos regulares se (e apenas quando) for necessário.

A Figura 15 mostra dois perfis de utilizador (“*Peter Smith*” e “*George Brown*”) e um estereótipo (“*Kunstliebhaber*” – amante de arte) na interface original do UMS. O perfil do utilizador *Peter Smith* está seleccionado, sendo possível visualizar a sua lista de interesses e valores associados à característica seleccionada.

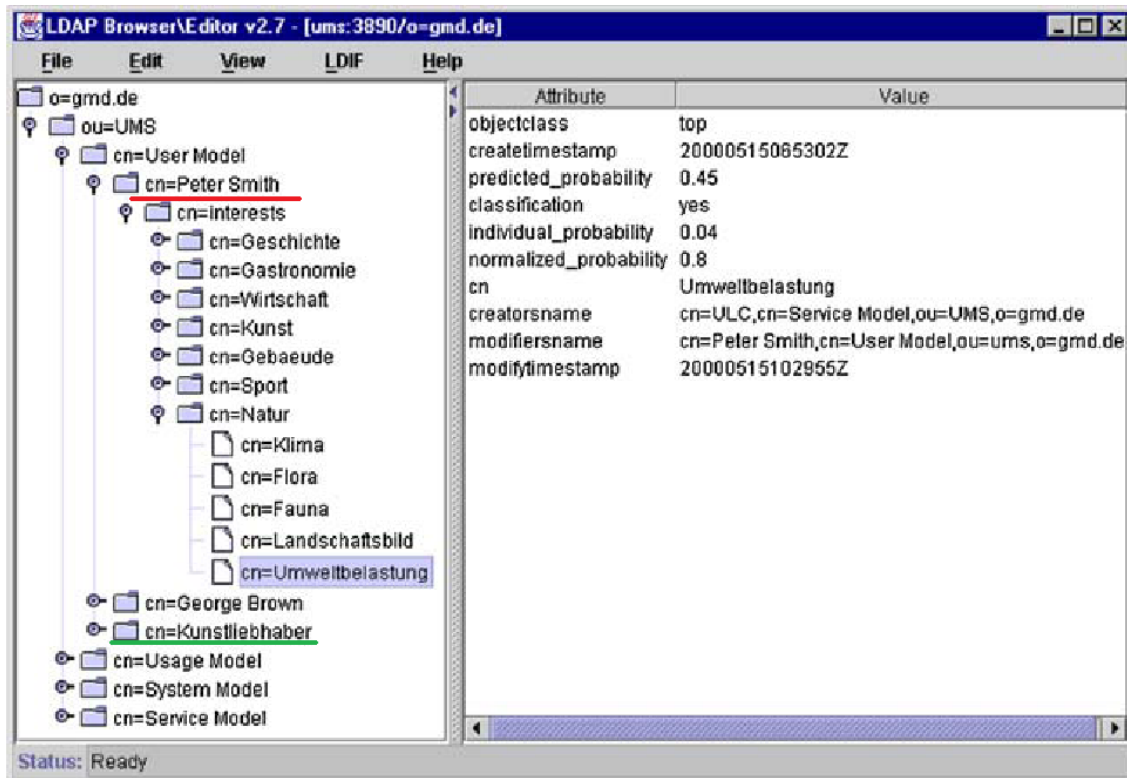


Figura 15 - Navegação LDAP no UMS (Kobsa & Fink, 2006)

Em (Kobsa & Fink, 2006) os autores explicam o funcionamento deste servidor com três componentes de modelação de utilizador distintos: aprendizagem de utilizador, aprendizagem de mentor e inferências de domínio.

A aprendizagem de utilizador infere os interesses e preferências do utilizador, de acordo com os dados de utilização, e actualiza o modelo do utilizador de acordo com as informações aprendidas.

A aprendizagem de mentor infere valores de características em falta, no modelo do utilizador, de acordo com os valores dos utilizadores com modelos semelhantes.

A aprendizagem de inferências tenta encontrar valores de características em falta, no modelo do utilizador, aplicando inferências do domínio próprio da aplicação aos dados recolhidos acerca do utilizador.

A comunicação das aplicações cliente com o UMS pode ser feita recorrendo a LDAP ou ODBC.

4.3.6 Generic User Modeling Server for Adaptive Web Systems

O *Generic User Modeling Server for Adaptive Web Systems* (GUMSAWS) é um servidor de modelos de utilizador genérico para aplicações web adaptativas (Zhang & Ghorbani, 2007). Trata-se de uma proposta de servidor central, que permite a várias aplicações concorrentes usarem o mesmo perfil de utilizador. É um sistema modular, que pode ser implementado em cenários distribuídos, dividido em quatro blocos ou módulos principais: repositório, gestor de modelos, descrição de modelos e gerador de informação, como ilustra a Figura 16.

O repositório mantém num sistema de base de dados a informação coligida pelos componentes que actuam sobre os perfis de utilizador e de grupo.

O gestor de modelos é responsável por garantir a correcta utilização dos modelos de utilizador, fornecendo métodos e formas de actuar sobre os mesmos (inserir, listar, ver, apagar e editar, perfis ou informação interna do perfil). Internamente, este módulo depende de dois componentes: o gestor de perfis de utilizador e o motor de inferências.

O gestor de perfis é responsável por tudo o que diz respeito à interacção das aplicações com o perfil do utilizador e pela inicialização do perfil, de acordo com o modelo de utilizador definido.

O motor de inferências é responsável por deduzir, de acordo com as fontes de informação disponíveis (perfis de utilizador, grupo, modelos de utilizador e modelo de grupo), dados em falta no perfil do utilizador, usando para tal algoritmos de inferência.

A descrição de modelos armazena num repositório de dados (que pode ou não ser coincidente com o módulo de repositório informação) especificações dos modelos de utilizador e dos grupos de utilizadores. Nestes modelos são descritos os factos, conhecimento (dependentes do domínio de aplicação) e as suas relações com os utilizadores e grupos.

O gerador de informação cria informação sobre os grupos e regras de associação para o gestor de modelos poder inferir valores para as propriedades sem valor (ou com valor por omissão) do perfil de utilizador. Este módulo contém dois componentes: o gestor de grupos e o gestor de associações. O primeiro associa utilizadores a grupos e o segundo define e gere as regras de associação de utilizadores a grupos.

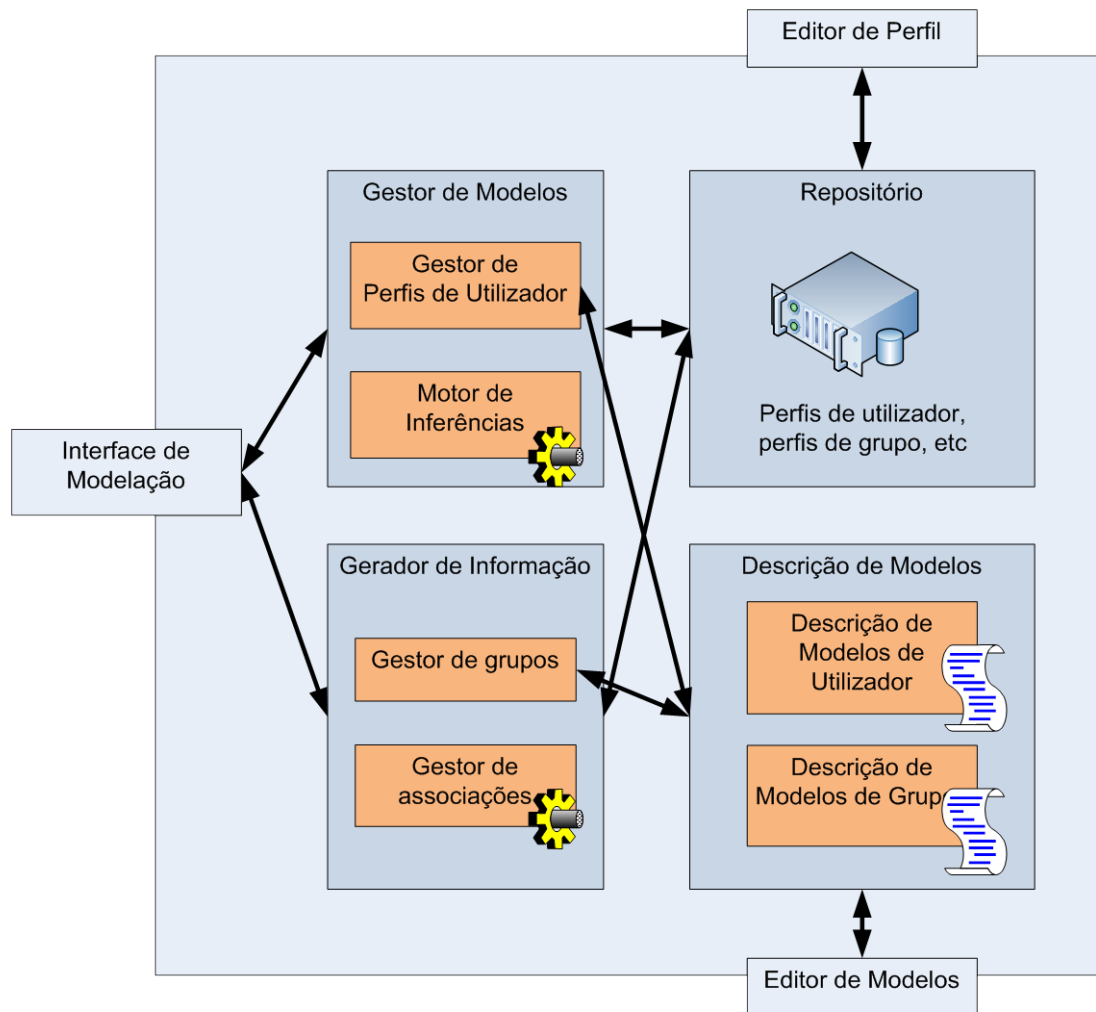


Figura 16 - Arquitectura do GUMSAWS

O GUMSAWS tem três pontos de interacção com aplicações: interface de modelação, editor de perfil e editor de modelos.

A interface de modelação recebe os pedidos sobre os utilizadores vindos de aplicações adaptativas e reencaminha-os para o módulo correcto.

O editor de modelos, permite acesso directo às descrições de modelos, permitindo a um administrador de sistema a (re)configuração de modelos de utilizador e de grupo, incluindo a definição dos valores por omissão para as características associadas aos modelos.

O editor de perfil acede directamente ao repositório de dados, permitindo aos utilizadores consultar e/ou alterar os dados mantidos sobre este.

4.3.7 Comparação dos sistemas de modelação

A Tabela 3 sumaria as principais características dos sistemas referenciados nesta secção.

Como se pretende que o perfil de utilizador possa ser usado por outras aplicações além do portal, de entre os sistemas apresentados, apenas os sistemas de tipo “servidor” poderiam ser usados neste caso particular. Destes, nenhum permite exactamente o que se pretende: um serviço independente da infra-estrutura de informação da organização, mas que mediante determinados mecanismos possa ter acesso à mesma, para definição de perfis de utilizador iniciais.

O sistema que se pretende deverá basear-se em permissões de acesso a operações e personalização por parte do utilizador, ao invés dos níveis de certeza e presunções dos sistemas apresentados. Assim, no capítulo 6 – “Sistema de gestão de perfis” será apresentada uma proposta alternativa que colmata as falhas indicadas. Pelos mesmos motivos, e porque o perfil de utilizador relaciona-se com o sistema de modelação, no capítulo 5.5 – “Integração do perfil de utilizador no portal” será proposto um modelo de perfil de utilizador distinto dos apresentados no estado de arte e que responda às necessidades deste sistema. Este deverá capacitar a personalização do acesso a serviços em ambiente corporativo, baseado em regras e permissões que poderão ser altamente dinâmicas. Para uma mais fácil gestão dos perfis de utilizador, estes terão por base um conjunto de perfis básicos. Os perfis básicos poderão assentar numa estrutura hierárquica de perfis, em que as permissões de um perfil básico “pai” são herdadas pelo perfil básico “filho”.

	GUMS	UMT	Tagus	Personis	UMS	GUMSAWS
Separação entre aplicação e modelação	Sim	Sim	Sim	Sim	Sim	Sim
Capacidade de inferência	Sim	Sim	Sim	Sim	Sim	Sim
Tipo de sistema	Concha	Concha	Concha	Servidor	Servidor	Servidor
Níveis de certeza	4 níveis	4 níveis	Níveis não fechados (vários)	Modelo probabilístico	Modelo probabilístico	Modelo probabilístico
Escrutínio	Não	Não	só programadores	Sim	Sim	Sim
Controlo de privacidade	Não	Não	Não	Sim	Sim	Sim
Perfil partilhado	Não	Não	Não	Sim	Sim	Sim
Características internas relevantes	Baseado em Prolog	Baseado em Prolog	Baseado em Prolog	<ul style="list-style-type: none"> • Base de dados relacional; • Interpretadores de inferências distintos 	Baseado em LDAP	Servidor Modular

Tabela 3 - Comparação de sistemas de modelação

4.4 Armazenamento de Perfis de Utilizador

Para que o uso de perfis de utilizador seja o mais eficaz possível, é necessário considerar as condicionantes de armazenamento, pesquisa e indexação dos mesmos. É, portanto, necessário identificar de que forma os modelos gerais de utilizador e os perfis de utilizador são armazenados.

Como referido por Kobsa e Fink (Kobsa & Fink, 2006), só a partir de 2002, é que se começou a dar importância às questões de armazenamento e performance nos sistemas de modelação de utilizador. De facto, um dos primeiros sistemas que fez destas preocupações um requisito, foi o servidor Personis. Até então, a maioria dos sistemas carregava para memória todo o perfil do utilizador, o que não é um cenário escalável para universos com potencialmente milhares de utilizadores em simultâneo.

Além dos sistemas de gestão de bases de dados relacionais tradicionais e da base de dados interna dos servidores LDAP, há também que não descurar a emergência do formato XML, não só para comunicação de informação, mas também para o seu armazenamento (Wugofski, 2000). Esta secção dedica-se ao estudo dos métodos que poderão ser usados para gerir informação armazenada em XML.

Um dos principais problemas quando se trabalha com informação armazenada em XML é a sua indexação e posterior recuperação por meio de pesquisa. De entre os repositórios de dados que aceitam formatos XML, podemos considerar três tipos distintos:

- Bases de dados relacionais que guardam todo o ficheiro como texto num campo da base de dados ou
- Bases de dados relacionais que extraem do XML schema (schema) (Wugofski, 2000) uma colecção de tabelas onde depois guardam a informação (Bourret; Corporation) – *XML Enabled Databases* (XenDB);
- Bases de dados de XML nativas – *Native XML Databases* (NXD), cada uma com a sua arquitectura interna, como explicado por Khin-Myo Win et. all (K. M. Win, 2003).

Há ainda uma quarta abordagem que consiste na utilização do sistema de ficheiros como repositório de documentos (filesystem databases). Muito tem sido o trabalho desenvolvido para melhorar os mecanismos de indexação e pesquisa de documentos XML (E. Bertino, 2004; Jason, Serge, Roy, Dallas, & Jennifer, 1997; Jungsuk Song, 2004; P. Rao, 2004; W. Zhang, 2004).

4.4.1 Bases de dados relacionais

Ao querer armazenar XML numa base de dados relacional, pode-se cair na tentação de considerar todo o documento XML como texto e, como tal, armazená-lo numa única

coluna. Como é óbvio, esta abordagem oferece muito pouco controlo sobre a informação contida no XML, pelo que será descurada.

As XEnDB, procuram mapear a estrutura de um documento XML num conjunto de tabelas relacionadas. De forma a ser o mais eficiente possível, estas podem basear-se no esquema (XSD ou DTD) que rege o XML ou, caso este não exista, na estrutura extraída dos documentos XML. A existência de um esquema pré-definido permite uma transformação de informação confiante de estar a ter em consideração todas as possibilidades (de acordo com o esquema). No entanto, no caso de não haver esquema associado, a base de dados relacional pode sofrer alterações quando novos documentos com elementos completamente novos sejam introduzidos. Mesmo no caso da existência prévia de um esquema da estrutura do XML e, dada a natureza dinâmica associada ao XML, é possível o esquema evoluir ao longo dos tempos, dificultando deste modo a tarefa da conversão de informação de XML para uma base relacional.

Estas bases de dados aceitam XML como entrada e produzem também XML como saída.

4.4.2 Bases de dados XML nativas

As NXD estão numa fase inicial de desenvolvimento (Kroenke, 2005; Mabanza, Chadwick, & Rao, 2006) e não podem ainda competir directamente com as bases de dados relacionais. Contudo, considerando a quantidade de pesquisas e aplicações que usam informação baseada em XML, não se pode, de forma alguma, descurar o facto do aumento de informação estruturada em XML estar a levar ao aumento da popularidade das NXD. As NXD são vistas por alguns como substitutos naturais para as bases de dados relacionais (Kroenke, 2005) mas, como refere o autor, ainda muito trabalho é necessário para tirar partido deste paradigma. As NXD têm a vantagem de toda a entrada, processamento, armazenamento e saída de informação ser em XML, não havendo lugar a conversões entre tipos de dados, o que, mostrando-se a tecnologia completamente capaz, poderá ser encarado como uma vantagem em termos de performance.

4.4.3 Sistemas de ficheiros

Tendo em consideração que a informação a ser armazenada tem uma perspectiva centrada no documento da informação, uma das soluções é armazenar os documentos XML directamente no sistema de ficheiros. Em sistemas de bases de dados tradicionais, esta

abordagem seria impensável, por carecer de mecanismos para assegurar a integridade da informação; os SGBDs têm ferramentas internas e automatizadas para gestão de transacções, cópias de segurança, aplicação de integridade, entre muitos outros. Esta solução não tem algumas das vantagens das bases de dados (como camadas de acesso a dados padronizadas), mas tem as vantagens que o próprio sistema de ficheiros oferece (escalabilidade e redundância facilmente configuráveis). Contudo, para poder ser eficaz em termos de pesquisas, necessita de um motor de pesquisa externo.

Como possibilidades para motores de indexação e pesquisa em ficheiros XML podem ser incluídos o Index Server da Microsoft, Lucene (Letecky, 2006) e o Google (DuCharme, 2004), entre outros. Já não se trata de sistemas de base de dados na verdadeira acepção da palavra, mas de sistemas dedicados a indexar e pesquisar sobre documentos. A operação de pesquisa é a principal actividade destes sistemas e o sincronismo entre repositórios de documentos e mapas de índices não lhes é exigido, podendo haver períodos de latência. Em sistemas de bases de dados, tais períodos seriam quase impensáveis, dada a sua natureza. Contudo, este compromisso pode não ser muito relevante em alguns cenários.

4.4.4 Comparação entre sistemas de armazenamento

A Tabela 4 sumaria as principais diferenças entre os sistemas de armazenamento referidos anteriormente.

O documento ida-e-volta (*Round-trip document*) explica a possibilidade de armazenar um documento XML e, mais tarde, recuperá-lo exactamente como fora introduzido, incluindo comentários e secções CDATA.

As NXDs são aparentemente a melhor opção, mas só após se terem afirmado em termos de performance e escalabilidade.

Dada a natural flexibilidade do XML, uma abordagem relacional pode ser usada se:

- Os XML schemas forem bem conhecidos (o que mantém a base de dados estável)
- A performance de introdução e saída de informação não for significativa (dado o impacto de transformar XML para relacional e vice-versa)
- A informação for actualizada com elevada frequência

	XEnDB	NXD	XML sistema de ficheiros
Actualização concorrential de informação	Maioritariamente baseado na totalidade do documento	Sim	Totalidade do documento
Introdução de XML	Transformação do XML para estrutura interna, ou introdução directa	Transformação do documento para estrutura relacional	Introdução directa
Recuperação de XML	Recuperação directa	Transformação de relacional para documento XML	Recuperação directa
Actualização de XML	XQuery/Xupdate	Query de linguagem relacional ou XQuery/XUupdate convertido	Substituição de ficheiro
Índices / Pesquisa XML	Sim	Funções para índices / pesquisa em BD relacional	Com motor externo
Documento ida-e-volta	Normalmente existe	Pode não existir	Sempre

Tabela 4 - Comparação entre os diferentes modelos de armazenamento

O XML em sistema de ficheiros pode ser uma má solução se a frequência de actualização for elevada, dado que necessita de abrir o documento, actualizar a informação e, finalmente, gravá-lo de volta. No entanto, a recuperação e armazenamento de informação serão rápidas, tão rápidas quanto o sistema de ficheiros o permitir. Será apenas necessário um motor externo para gestão de indexação e pesquisa de informação.

5 O Portal my.ua

Nos capítulos anteriores foram apresentadas tecnologias de suporte e sistemas análogos ao que se pretende implementar. No entanto, nenhuma das soluções apresentadas consegue responder de forma cabal ao que se pretende como propósito do sistema. Assim, é necessário desenvolver um sistema novo, que responda ao pedido original e que tenha em consideração as boas práticas e lições recolhidas da análise dos sistemas apresentados no estado da arte.

Do ponto de vista de interação com o utilizador, será necessário desenvolver um sistema de portal web que seja independente (do ponto de vista de desenvolvimento) dos componentes de informação e de operações que irá apresentar (capítulo 5). Uma vez que o perfil do utilizador ditará a informação e operações disponíveis para cada caso, o portal deverá ser capaz de, dinamicamente, construir a página final que irá apresentar ao utilizador.

Do ponto de vista de sistema de informação associado ao portal, será necessário desenvolver um sistema de gestão de perfis e um modelo de perfil de utilizador que permita definir para cada utilizador, quais as permissões que lhe são aplicáveis num dado momento (capítulo 6). Estas definições poderão ser feitas nos perfis básicos que depois serão aplicados ao caso concreto do utilizador. As páginas do portal serão construídas com base na informação constante no perfil do utilizador. As páginas serão constituídas por componentes criados para o efeito e apresentados no capítulo 7. Cabe aos componentes a apresentação e gestão (se aplicável) da informação constante nas diversas fontes de dados da Universidade.

5.1 Introdução

O protótipo desenvolvido, com o nome my.ua, visa ser uma plataforma integradora de sistemas e serviços existentes na Universidade que de alguma forma se relacionem com um utilizador final. Trata-se de um CPAD, uma vez que é completamente personalizável: cada utilizador pode ter uma interface à sua medida. Trata-se, portanto, de um portal dinâmico com interface reconfigurável em função das preferências e privilégios do utilizador autenticado. Isto significa que, por exemplo, um estudante autenticado no portal poderá ter uma visão completamente diferente de um docente (C. Teixeira, Pinto, & Santos, 2005).

O propósito do sistema é apresentado na Figura 17.

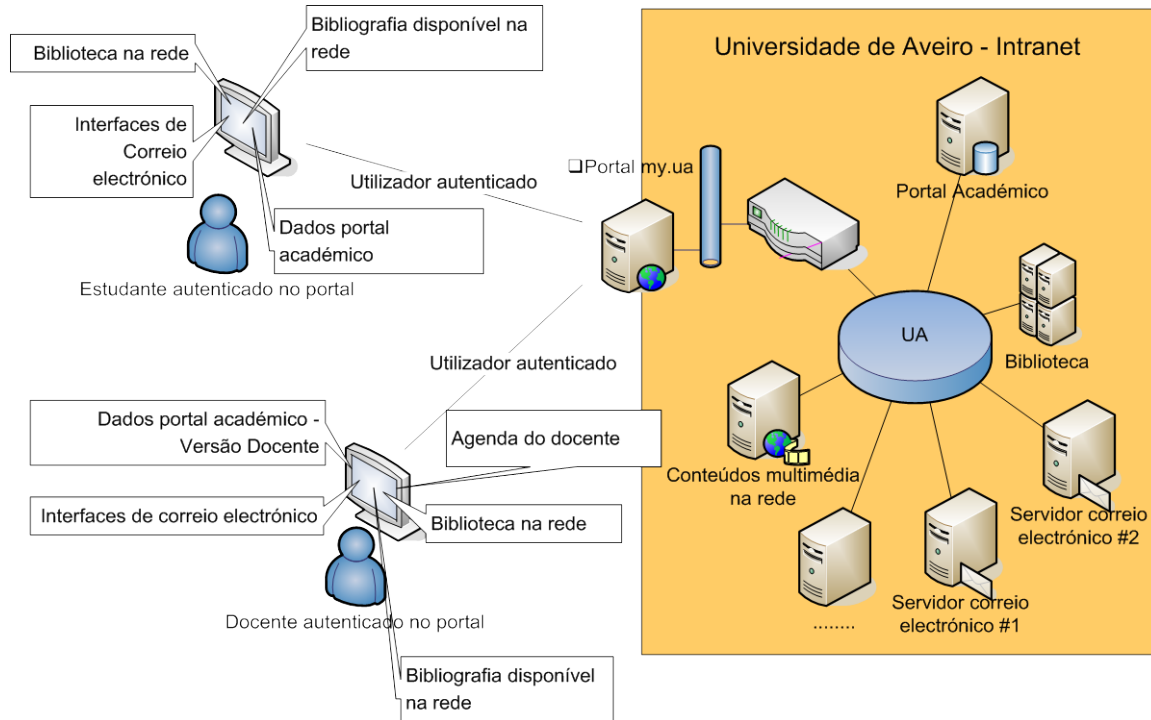


Figura 17 – Vista prática do portal como integrador de informação

Para que o sistema cumpra o seu propósito, a interface deverá ser capaz de se adaptar ao utilizador autenticado. Não se trata apenas de modificar os dados na interface, mas a própria estrutura da interface poderá ser afectada: mais blocos informativos para um determinado utilizador, blocos comuns, blocos únicos para um determinado conjunto de utilizadores.

Para se poder satisfazer este requisito, a interface global não pode ser construída durante o modo de desenvolvimento do portal. Apenas o esqueleto do sistema (a plataforma de portal) pode ser completamente definida. A partir desse ponto, será necessário que, cada vez que um utilizador deseje consultar uma determinada página, esta seja construída especificamente para ele, com todos os componentes e informação correcta, tendo em consideração as preferências demonstradas anteriormente pelo utilizador.

5.2 Arquitectura

A arquitectura base do sistema foi apresentada em (C. Teixeira et al., 2005). Do ponto de vista funcional, o sistema é composto por quatro blocos principais: autenticação, validação, perfil e interface (Figura 18).

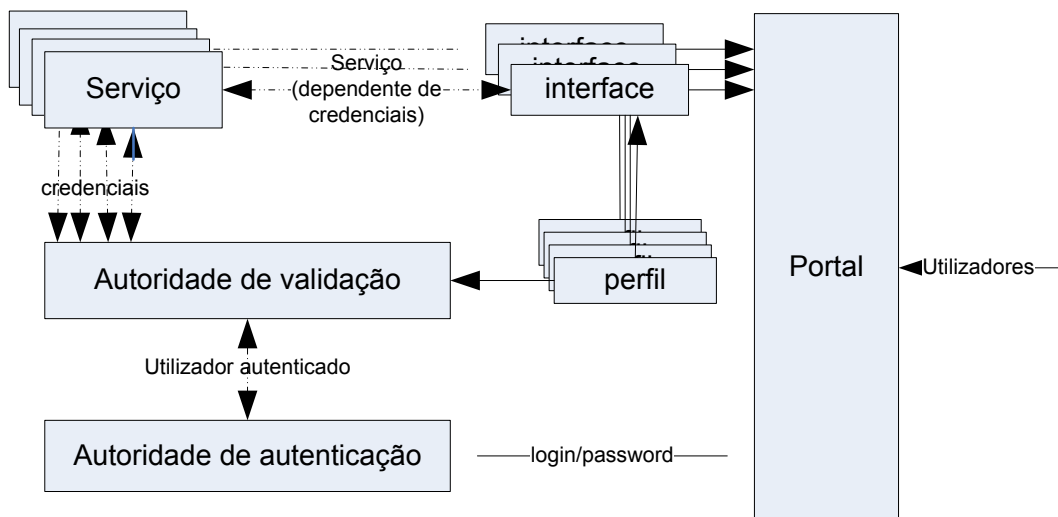


Figura 18 - Arquitectura do sistema

O bloco de autenticação é o responsável por garantir que as credenciais submetidas por um utilizador estão ou não correctas. Caso estejam correctas, então a identificação do utilizador é transmitida ao sistema (garante-se desta forma a validade do utilizador). O bloco de autorização verifica a legitimidade do acesso por parte do utilizador, a um dado recurso. Apesar de já ter sido autenticado pelo sistema, este poderá não ter permissões para aceder a um determinado recurso. A informação relativa aos privilégios de um utilizador, sobre um determinado recurso, está armazenada no perfil do utilizador. Os perfis dos utilizadores são geridos externamente ao portal, usando para tal um sistema de gestão de perfis desenvolvido. Analisando o perfil do utilizador, o portal verifica quais as interfaces com sistemas a que este tem acesso, em que moldes e como é que estas lhe deverão ser mostradas.

Depois de compilar as interfaces com sistemas, a plataforma constrói dinamicamente a página que irá ser mostrada ao utilizador. A interface gerada possibilitará assim, ao utilizador, interagir com os serviços a que tem acesso naquele momento.

Em cada componente, a interface gráfica, os dados mostrados e o modo de acesso aos mesmos, é da responsabilidade do serviço que o desenvolveu. O portal serve apenas como suporte lógico e meio integrador dos diferentes serviços.

5.3 Modelo de Dados

Do ponto de vista de dados, o sistema armazena internamente informação específica sobre a interação do utilizador com o portal. Garante-se desta forma que a maioria da

informação sobre o utilizador é acedida em tempo real, o que evita questões de actualização de informação. Do mesmo modo, sempre que é necessário (e possível) actualizar informação existente noutros sistemas, esta actualização é efectuada da forma mais directa possível. A actualização de informação implica mecanismos de acesso controlados pelo serviço em causa, que deverão validar a operação pretendida.

O portal armazena informação sobre configurações específicas: dados de acesso a contas de correio electrónico não associadas à universidade, endereços de canais noticiosos, informação sobre imagem do utilizador (avatar), dados de agenda institucional, entre outros.

Apesar de se tratar de informação bastante especializada, para uso por parte de alguns componentes que poderão não estar acessíveis a todos os utilizadores, a razão que levou a que fosse o próprio portal a armazenar esta informação, prende-se com uma questão de facilidade de implementação. Como estes componentes foram desenvolvidos em paralelo com a própria estrutura do portal e, por uma questão de agilização de desenvolvimento, optou-se pela sua inclusão no modelo de dados intrínseco ao portal. No entanto, poderiam ter sido completamente dissociados, visto serem modulares e independentes.

Se os componentes como o leitor de correio electrónico, a agenda, o leitor de notícias, etc., forem analisados de forma independente do portal, então este efectivamente não armazena nenhuma informação sobre o utilizador, visto que o próprio perfil do utilizador é gerido externamente ao portal.

Por questões relacionadas exclusivamente com a performance do sistema, componentes como a agenda armazenam internamente dados que à partida não sofrem mudanças frequentes, como é o caso da lista de disciplinas associadas a alunos e docentes (tipicamente duas mudanças por ano). No entanto, de forma a garantir que as eventuais alterações são efectivamente registadas, é da responsabilidade do componente verificar a actualidade da informação armazenada localmente a intervalos regulares.

5.4 Integração de sistemas e serviços

Como nesta secção se pretende fazer a descrição da interacção dos vários serviços, convém, de modo a que se percebam melhor os fundamentos, fazer uma abordagem prévia a arquitecturas orientadas a serviços.

5.4.1 Arquitecturas orientadas a serviços

A orientação a serviços não deverá ser vista como uma tecnologia, mas sim como um paradigma de organização de sistemas (Margaria, 2007). Trata-se de um modelo arquitectural cuja meta é aumentar a eficiência, agilidade e produtividade de uma empresa, focando em serviços os meios pelos quais é representada a lógica de negócio (Erl, 2008). Segundo (Oasis-open.org, 2008) um sistema baseado em SOA é semelhante a um ecossistema: um espaço onde pessoas, máquinas e serviços interagem para atingir objectivos particulares de cada interveniente e/ou objectivos gerais.

Em (Balzer, 2004), o autor define princípios gerais de boas práticas para SOA, alguns dos quais adaptados de programação orientada a objectos:

- Reusabilidade, granularidade, modularidade, composabilidade e componentização –ou seja, os serviços disponibilizados poderão ser potencialmente reutilizados noutros cenários, sendo específicos o suficiente para a resolução do problema; deverão ser auto-suficientes, no sentido em que o serviço possa ser entendido como um módulo, que processa de forma independente e autónoma uma dada instrução; deverão ainda suportar composição de serviços mais elaborados, usando para tal os próprios serviços como componentes orquestrados ou encadeados numa dada sequência.
- Adesão a normas – permite maximizar as potencialidades de interoperabilidade do serviço com outros.
- Identificação e categorização de serviços, assim como provisionamento, entrega, monitorização e acompanhamento dos mesmos – os serviços em produção deverão ser correctamente identificados e categorizados, de forma a facilitar a tarefa de pesquisa e retorno do serviço correcto para uma dada operação. Além disso, estes deverão ser vigiados de perto pelos responsáveis, a fim de detectar eventuais anomalias.

Além destes princípios gerais, o autor define ainda princípios específicos da arquitectura:

- Encapsulamento – este paradigma transforma todo o serviço numa “caixa negra”, onde o importante, do ponto de vista de utilização do mesmo, é saber quais os dados esperados à entrada, e qual o resultado esperado à saída.

- Separação da camada de negócio da tecnologia de base – uma vez que o serviço está abstraído numa caixa negra, é em qualquer instante possível alterar a tecnologia base, mantendo a lógica de negócio.
- Implementação única e vista global de componentes – este princípio pode ser visto como um corolário dos princípios gerais, visto que, se os serviços devem ser reutilizados em serviços cada vez mais complexos, então estes deverão ter implementação única na empresa/meio onde serão usados, estando ao mesmo tempo visíveis e disponíveis para composição de outros serviços.
- Reaproveitamento de lógicas existentes sempre que possível – se for possível adaptar um serviço para ser inter-operável, quando até então servia apenas uma aplicação, então esta abordagem ganha todo o tempo de desenvolvimento da fase de operação com uma única aplicação, conseguindo disponibilizar um serviço maduro, já bastante cimentado na lógica do negócio e com provas dadas.
- Gestão do ciclo de vida – tal como em qualquer sistema em produção, será necessário gerir o ciclo de vida do serviço.
- Uso eficiente de recursos do sistema – uma vez que cenários de interoperabilidade conferem um potencial de maior utilização dos serviços disponibilizados, estes deverão acautelar questões de desempenho e escalabilidade da operação para casos de elevada utilização.
- Maturidade e desempenho do serviço – os serviços disponibilizados deverão estar já num ciclo de vida de produto que permita garantir a sua robustez e desempenho.

Um serviço é uma unidade autónoma não dependente da plataforma, que pode ser publicado, descoberto e agregado a outros de novas formas (Papazoglou, Traverso, Dustdar, & Leymann, 2007). Um serviço não necessita de contextos ou estados de outros serviços para satisfazer a sua função (poderá, contudo, invocar outros serviços como parte do seu processo de execução).

Actualmente, a recorrência a serviços web como meio de comunicação entre serviços é o mais comum neste paradigma, principalmente pelo seu contributo para a transparência do serviço subjacente.

Essencialmente, qualquer componente lógico de uma aplicação pode ser transformado num serviço usável em rede. Estes serviços podem ser orquestrados e coordenados de diversas

formas, de acordo com a necessidade e especificidade da tarefa que seja necessário realizar.

A orquestração de serviços é conseguida recorrendo a linguagens de alto nível que permitem especificar a forma de encadeamento dos diferentes serviços. *Business Process Execution Language* (BPEL (OASIS, 2007)) é a linguagem padrão usada nesta definição.

5.4.2 A arquitectura proposta

A especificação original do portal previa, por questões relacionadas com a “segurança e confiança” do sistema de informação da Universidade de Aveiro, um sistema apenas de apresentação de informação proveniente de outros sistemas. Todas as operações de actualização e/ou introdução de informação relacionada com o utilizador (à excepção do correio electrónico), teriam que ser efectuadas usando os sistemas já existentes para o efeito. O portal seria uma montra de serviços e informação, sem possibilidade de “mexer” na mesma. De acordo com esta exigência, foi feito um levantamento dos diferentes sistemas existentes na Universidade, assim como da forma de aceder à informação gerida por estes. A Figura 19 representa esse estudo. À data, nos serviços assinalados, já tinham sido projectadas remodelações internas (ou eram serviços para ser implementados de raiz) de forma a garantir uma possível interoperabilidade.

Desta figura ressalta o facto de não haver ligações entre os diferentes sistemas, ou seja, o sistema de informação existente era na verdade composto por diversas ilhas de informação (C. Teixeira, Pinto, & Martins, 2008a), cada uma com o seu conjunto de nomes de utilizador e palavras passe, regras e políticas de segurança e de acesso.

Infra-estrutura para portal internet integrador de serviços

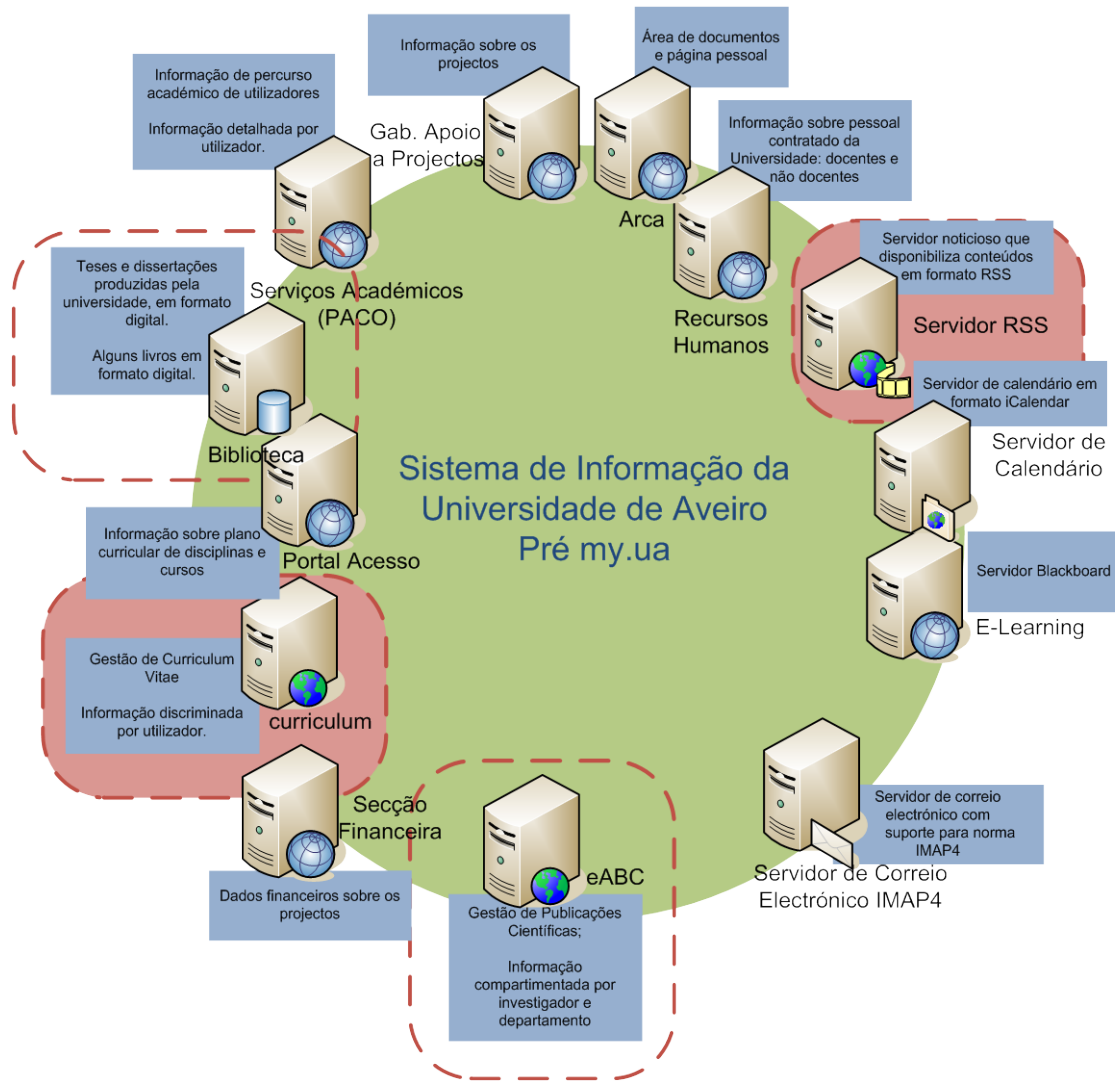


Figura 19 - Serviços de informação existentes na Universidade.

Além disso, verificou-se que a maioria dos sistemas não possuía qualquer mecanismo para permitir a troca da informação de que era responsável, com sistemas externos. Assim, o levantamento que deveria resultar em cenários concretos de interoperabilidade com o portal, acabou por se transformar num mapa da informação existente. Neste mapa faltavam os “trilhos” ou pontes para a comunicação. Foi então necessário avaliar qual a informação que estava em cada sistema, que nunca tinha sido projectada para ser utilizada noutra lugar que não o próprio serviço e definir qual o modo de possibilitar a partilha dessa informação. O acesso a esta informação deveria obedecer aos pressupostos iniciais do projecto: acesso apenas para leitura e a apenas parte da informação existente no sistema.

Os sistemas analisados podem ser agrupados em função da sua importância para o regular funcionamento da universidade: os vitais e os secundários. Os sistemas vitais têm, em

regra, um longo período de desenvolvimento e aperfeiçoamento dentro da universidade, e não podem parar para repensar a sua forma. Assim, têm que se ir adaptando às mudanças impostas pela própria universidade. São exemplos, o Portal Académico (PACO), os Serviços Académicos, os Recursos Humanos e os Recursos Financeiros.

Os sistemas secundários poderão também ser sistemas robustos, dado o seu ciclo longo de desenvolvimento. Contudo, são sistemas mais tolerantes a interrupções de serviço, com vista a melhorias de arquitectura e/ou funcionalidades. São exemplos, o Curriculum Online e o eABC. Em ambos os casos, estes sistemas já estavam assinalados para ser reestruturados (no caso do eABC) ou implementados (no caso do Curriculum). A integração (ou possível integração) destes sistemas com outros pôde ser acautelada praticamente de raiz, dadas as novas implementações.

Contudo, nos restantes casos não era de todo possível reconstruir o sistema. A solução passou por integrá-los como se de sistemas legados se tratassem (apesar de em muitos casos o seu desenvolvimento se manter). Esta foi aliás, a regra para acesso a qualquer sistema de informação: tratá-lo como um sistema legado, mantendo ao máximo a sua “insularidade” mas garantindo “pontes” de comunicação, acordadas entre os responsáveis pela manutenção do serviço e os responsáveis do my.ua, visto ser este o maior interessado. Importa referir que estas pontes poderão servir não apenas o portal, mas também outros sistemas.

Para uma correcta definição de arquitectura de serviços, foi necessária uma identificação unívoca dos diversos participantes, o que implica conseguir fazer corresponder os diferentes nomes de utilizador usados pelos diversos serviços, convergindo desta forma para uma identificação única de pessoas na universidade. Este passo foi o mais complexo de orquestrar dentro da universidade, dado o conjunto alargado de sistemas envolvidos. A solução passou pela criação de um Registo Central de Utilizadores (RCU), que é responsável por gerir credenciais e dados pessoais, bem como a troca de mensagens entre serviços.

Depois de acertados os pormenores de implementação do RCU, chegou o momento de analisar o levantamento de informação feito inicialmente e verificar quais os sistemas a integrar no portal numa fase inicial.

Todos os sistemas referenciados são passíveis de ser integrados, mas para demonstração de conceito e protótipo inicial, será dado destaque a:

- Dados académicos (PACO),
- Dados sobre cursos e disciplinas (Acesso),
- Correio electrónico,
- Sítios noticiosos externos e
- Áreas de documentos (ARCA).

Além destes, os seguintes sistemas, em fase de desenvolvimento simultânea ao portal, foram projectados de forma a poderem ser facilmente integrados:

- uaOnline (sítio noticioso interno),
- Serviço de notificações de utilizadores,
- eABC e
- Curriculum.

A comunicação entre serviços foi definida numa base SOA, com recurso a web services. O uso de web services potencia a abstracção do sistema e a valorização do serviço prestado, abrindo novas possibilidades de cruzamento de informação oriunda de sistemas distintos. Em casos particulares será usado RSS.

5.5 Integração do perfil de utilizador no portal

Após a definição de serviços e sistemas que irão interagir com o portal numa primeira fase, resta definir a lógica subjacente à interface que é apresentada ao utilizador, nomeadamente, ao nível da personalização do sistema. Deste modo, é necessário definir como é que após validação, aparece a informação do utilizador e qual a sequência de operações e que sistemas interagem no processo.

Tal como mencionado anteriormente, a informação de personalização do utilizador será registada num perfil de utilizador. Esta secção terá por base o trabalho descrito em (C. Teixeira, Pinto, & Martins, 2006; C. Teixeira, Pinto, & Martins, 2008c).

5.5.1 Elementos base

Cada elemento do perfil de utilizador está definido internamente numa abordagem em três camadas ou níveis distintos: Identificação, Validação e Interface.

A camada de **identificação** é responsável por claramente identificar o elemento. Habitualmente é composta por um identificador único e uma descrição textual.

Na camada de **validação** cada elemento pode estar sujeito a validação temporal e/ou validação por regras. A validação temporal pode referir, quer intervalos absolutos (exemplo: a inscrição de alunos em turmas práticas está disponível do dia 15 ao dia 21 de Setembro em cada ano), quer relativos (exemplo: acessos por tempo limitado – após ser concedida a autorização, esta vale por x dias). A validação pode também ser exigida por meio de regras formais, definidas para o elemento e aplicadas ao utilizador em concreto.

A camada de interface é a parte mais prática/operacional do elemento, definindo o que este realmente é/faz.

Os elementos chave do sistema de gestão de perfis são “Serviço”, “Operação”, “Contexto” e “Tema”.

5.5.1.1 Serviço

Trata-se de um elemento organizacional que está directamente ligado ao serviço físico (como os serviços académicas da universidade).

5.5.1.2 Operação

Este é elemento base dos perfis de utilizador. Representa uma operação que um utilizador pode executar (exemplo: ver correio electrónico), sendo que o seu propósito não está confinado ao uso exclusivo no portal. Cada operação é definida e suportada por um serviço específico, podendo cada serviço disponibilizar a quantidade de operações que entender, visto que contribui não só para o aumento de visibilidade do serviço, mas também para o sucesso de todo o sistema. Cada serviço pode definir as operações que entende, sendo da sua responsabilidade os contextos em que uma operação pode estar associada, e em que moldes e condições (validações temporais e formais).

A título de exemplo, pode-se imaginar um serviço que forneça acesso a um cliente de correio electrónico que queira disponibilizar dois conjuntos de funcionalidades: “ler/listar correio” e “ler/listar/enviar correio”.

Neste caso, o serviço deverá disponibilizar duas operações distintas, que serão associadas a contextos. Estes, por seu turno, estarão inseridos em perfis básicos. A Figura 20 e Figura 21 mostram o diagrama de classe para a operação (OperationType).

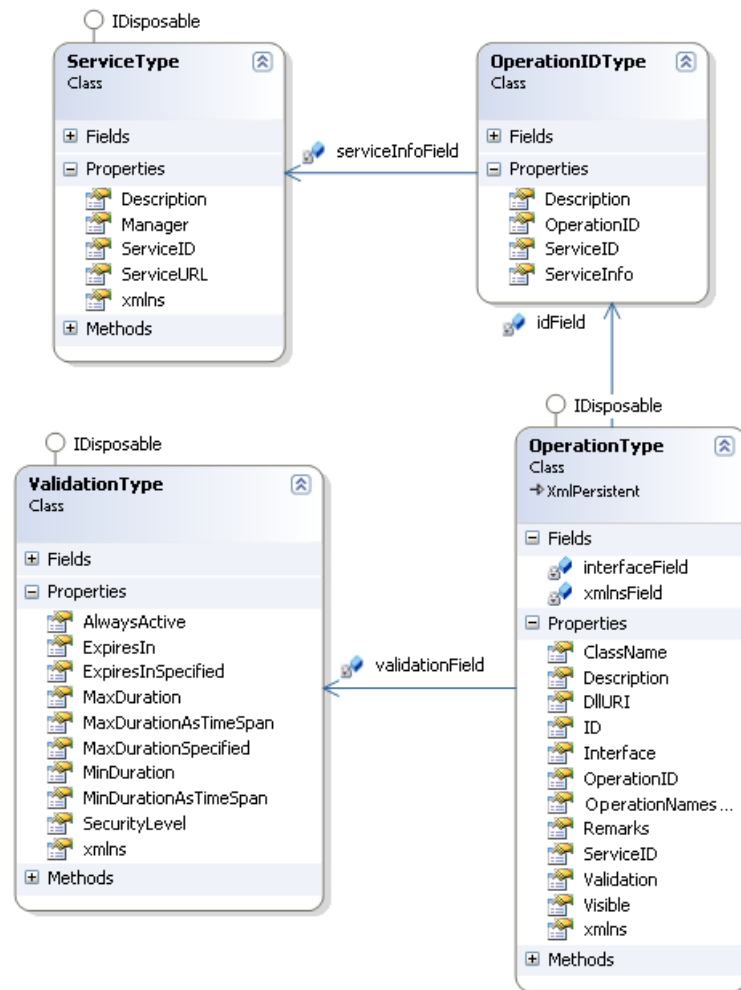


Figura 20 - Diagrama de classes do elemento Operação (parte 1)

Numa operação associada ao portal, a secção de identificação define a identidade da operação (identificador, serviço e pessoa responsável). A secção de validação é semelhante à secção de validação definida anteriormente. A interface é a parte visível da interacção entre o portal, o utilizador e o sistema de informação; a interface é definida pelo serviço e é específica para cada operação. Note-se que este elemento define, não as questões gráficas associadas à sua representação, mas os elementos requeridos. No caso do cenário do portal,

cada operação é compilada como uma *Dynamic Link Library* (DLL) que é colocada no servidor. Este objecto é definido pelo nome da DLL, o espaço de nomes da classe e o nome da classe (possibilitando desta forma que cada serviço forneça uma única DLL com várias operações). A DLL, por sua vez, contém as informações sobre uma correcta visualização na página.

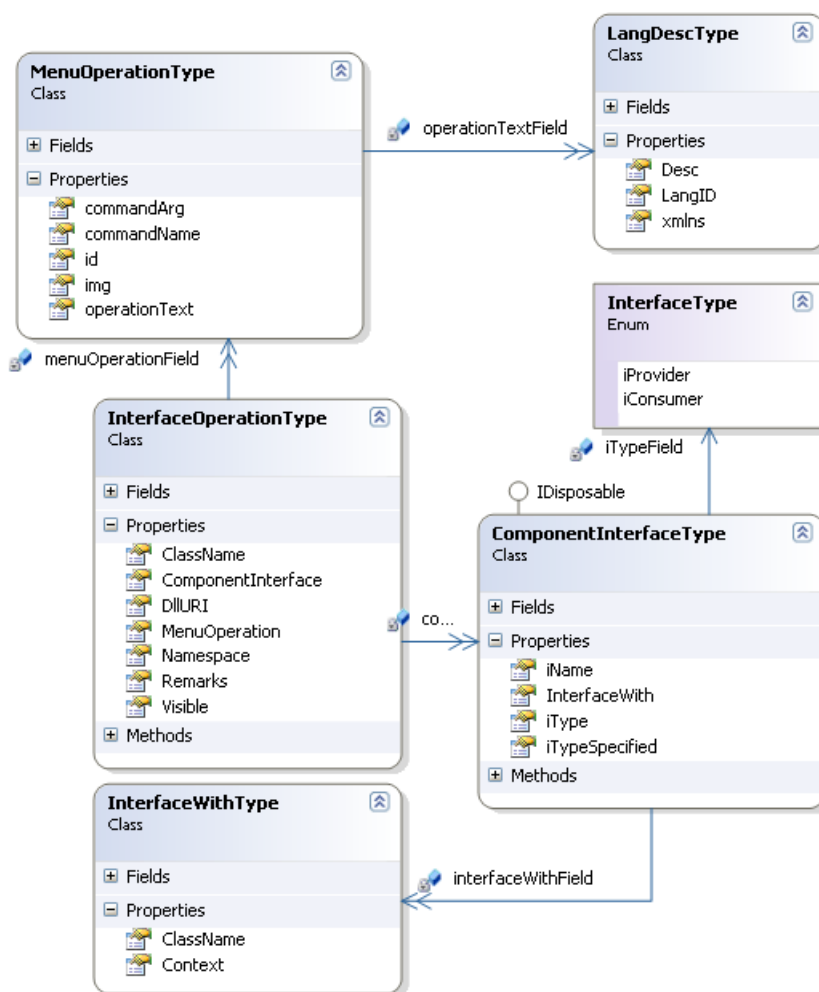


Figura 21 - Diagrama de classes do elemento Operação (parte 2)

Sendo este um elemento fundamental do sistema, necessita de uma explicação mais aprofundada de algumas das suas propriedades. Tendo sido definido para sistemas genéricos, a interface da operação, além do já mencionado, permite a especificação de entradas de menu que podem ser usadas pelo sistema anfitrião (neste caso o portal) para dinamicamente apresentar uma ligação para a dada operação. Esta ligação pode ser definida como texto simples, localizado ou mesmo como uma imagem, e pode conter informação mais detalhada, que deverá ser passada do sistema para a operação no

momento da sua selecção (como o nome de um comando e um argumento). Uma operação pode declarar-se como sendo componente de interface (ComponentInterfaceType).

A componente de interface define a possibilidade de uma operação comunicar com outras segundo um formato pré-estabelecido. Esta interface é declarada pela operação como sendo consumidor ou fornecedor, tendo como referencial o fluxo de informação. A componente de interface é definida programaticamente como uma interface em programação orientada a objectos. Interface essa que a operação declara respeitar. Deste modo, operações que estejam em execução simultânea no sistema, podem comunicar entre si, sem necessitarem de programar canais de comunicação específicos.

5.5.1.3 Contexto

Um contexto é um conjunto de operações de diferentes serviços que estão relacionados com o mesmo propósito (exemplo: “Contexto de mensagens e notificações” diz respeito a operações relacionadas com a notificação do utilizador). Um perfil base é composto por diversos contextos.

Num contexto que seja usado pelo portal, a secção de interface diz respeito ao modo como as diferentes operações são organizadas no ecrã do utilizador. Nesse caso, todas as operações definidas num mesmo contexto serão apresentadas numa única página, dando uma visão integrada dos serviços correspondentes.

Neste elemento, a parte mais importante é a interface, definindo-se uma nova subsecção: a “parte do contexto”. A parte do contexto é meramente uma unidade de arranjo gráfico que define como as operações são apresentadas na página. Pode-se pensar nesta subsecção como uma grelha virtual, que indica em que linha(s) e coluna(s) da interface web é que uma dada operação será mostrada. Em cada parte do contexto são definidas as operações que serão mostradas nesse local, na interface, por ordem de declaração.

5.5.1.4 Tema

Dado que os perfis serão usados num portal, o elemento tema permite uma personalização semi-automática. Cada perfil básico poderá incluir na sua informação vários temas. A lista de temas terá por base um thesaurus. Os utilizadores poderão também adicionar temas a partir de um dicionário. Com base nos temas, as operações poderão configurar informação

preferencial a mostrar ao utilizador (exemplo: Tema “desportos radicais” pode direccionar uma operação de leitura de notícias, a mostrar também, notícias relacionadas com desportos radicais).

5.5.2 Definição de perfil básico

Em (C. Teixeira et al., 2006) propomos uma abordagem estruturada ao perfil do utilizador, em que cada perfil resulta da agregação de vários perfis básicos. Estas definições de perfis básicos (*Basic Profile Definitions* – BPDs) podem ser vistas como modelos transaccionais, uma vez que definem o que um utilizador poderá ser (ou fazer) se tiver este BPD. Tendo em consideração o âmbito de aplicação dos perfis, cada perfil básico contém informação sobre o que o utilizador pode fazer no sistema de informação da universidade e onde é que os blocos informativos (as operações) serão colocados inicialmente na sua interface.

Esta abordagem é diferente da abordagem defendida no estado de arte de modelação de utilizadores. Neste caso, não estamos a definir interfaces adaptáveis para conteúdos ou produtos, estamos sim a utilizar a modelação de utilizadores para personalizar o acesso a serviços num ambiente corporativo, baseado em definições de funções e regras de permissões dinâmicas. Ainda assim, podem-se traçar alguns paralelismos entre os perfis básicos e estereótipos de previsão (Liliana & Anna, 2000): a parte do perfil assemelha-se ao processo formal que definimos e que contribui na atribuição de perfis básicos; e a parte de previsão é o perfil básico, visto que ambos descrevem características de utilizadores (acesso a operações, neste caso).

Cada BPD pode herdar características de outro BPD, o que significa que o novo BPD contém as permissões e validações do perfil pai juntamente com as suas próprias regras. Como possíveis BPD de nível 0 (sem pai) podem-se considerar, a título de exemplo: “estudante”, “docente”, “administrativo” e “imprensa”.

Para uma melhor compreensão do perfil de utilizador, considere-se o caso do perfil de um estudante. Por definição, cada estudante tem no seu perfil o BPD “estudante”. Este BPD define uma série de contextos, como “infra-estruturas de rede”, “PACO”, “Pesquisa e Conhecimento”, entre outros. O Contexto “infra-estruturas de rede” agrega várias operações, tais como, um cliente de correio electrónico integrado no portal, espaço num servidor central para documentos pessoais, ...

Tal como os restantes elementos, cada perfil básico é composto por três níveis distintos, como ilustrado na Figura 22.

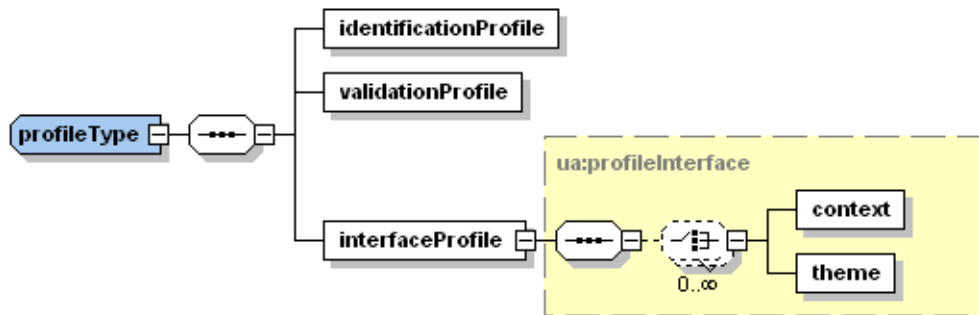


Figura 22- Esquema de perfil básico

Na identificação, cada perfil básico tem um identificador único, uma descrição textual e um identificador de perfil herdado (caso exista). Na validação, cada perfil básico pode ter uma validade temporal e uma menção ao gestor do perfil básico (quem pode alterar o mesmo). A validade temporal pode ser absoluta ou relativa. Finalmente, a interface contém informação mínima (apenas a parte de identificação) de elementos temáticos e de contextos.

5.5.3 Perfil de utilizador

A abordagem ao perfil de utilizador é relacionada com o serviço e, é directamente dependente da condição deste na organização (neste caso em particular, na universidade). Estas condicionantes não podem ser modificadas pelo utilizador; cabe a um conjunto de processos formais na organização revogar e dar permissões.

Na sua primeira visita ao sistema, o perfil do utilizador será simplesmente um conjunto de perfis básicos aplicados ao seu caso particular. Contudo, o utilizador tem a possibilidade de personalizar todo o portal, incluindo definições das operações a que tem acesso no momento. Além disso, cada operação tem acesso ao perfil do utilizador e pode adaptar a sua informação de acordo com os gostos e preferências expressos neste.

A Figura 23 representa o modelo associado ao perfil de utilizador completo.

Cada perfil de utilizador é praticamente único, visto que se trata da agregação dos privilégios definidos pelo conjunto de perfis básicos, em conjunto com a personalização do utilizador. Uma característica deste modelo é que as permissões são sempre positivas, significando que não pode haver instruções como “se o utilizador tiver esta autorização,

então perde acesso ao sistema ‘X’”. Em termos de actualizações, uma actualização efectuada pelo sistema de informação (ao perfil de utilizador) significa uma renovação ou revogação temporal, a um dado elemento. Deste modo, o perfil de utilizador poderá ser visto como um registo histórico de permissões e acessos do utilizador, além de conter elementos personalizados e características actuais.

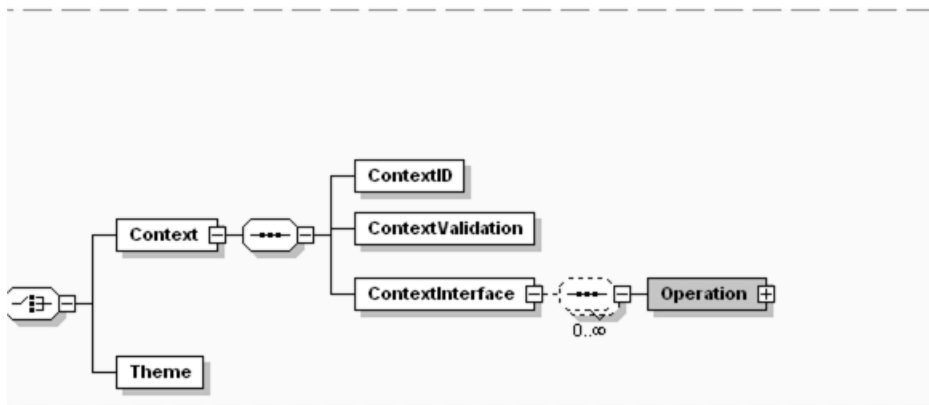
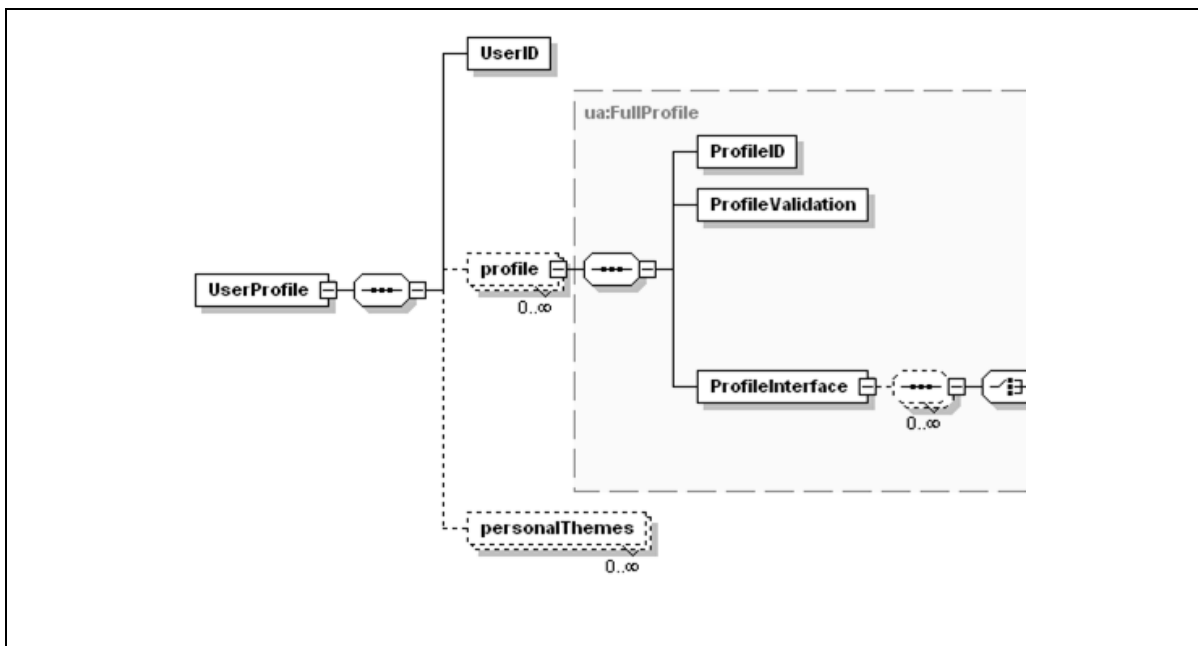


Figura 23 - Esquema completo do perfil de utilizador

Por uma questão de personalização, no portal desenvolvido, os utilizadores podem configurar quais os locais na interface onde pretendam visualizar algumas das operações (dentro de certos limites e apenas para os casos em que as operações permitem estas

movimentações). Do mesmo modo, podem desactivar operações, se estas assim o permitirem.

5.5.4 Alterações ao perfil de utilizador

Como referido anteriormente, o utilizador só pode alterar parte do seu perfil, sendo da responsabilidade da organização, alterar a restante informação. Tipicamente, existem dois tipos de actualizações de perfil de utilizador:

- Actualização directa – sempre que um processo formal actualiza um perfil de utilizador específico (revogando ou emitindo autorizações); sempre que um utilizador altera o seu perfil; sempre que uma aplicação ou serviço altera um perfil de um utilizador específico;
- Actualização indirecta – sempre que um processo formal actualiza um perfil básico, contexto ou operação (alterando regras de segurança, adicionando ou removendo autorizações). Quando este processo ocorre, é necessário aceder a todos os perfis de utilizador que contenham os perfis básicos que sofreram alterações e agir de acordo com as novas definições dos mesmos.

A actualização indirecta poderá parecer complexa, mas agiliza o processo de verificação de informação do utilizador, uma vez que se garante que o seu perfil está sempre actualizado de acordo com as orientações da organização.

6 Sistema de gestão de perfis

Como referido anteriormente, os sistemas de modelação apresentados não respondem aos requisitos para este projecto. Assim, neste capítulo será apresentada uma solução baseada na gestão de permissões de acesso a operações e personalização por parte do utilizador. Esta solução inova em relação aos sistemas estudados anteriormente, por ser uma abordagem à gestão de perfis de utilizador assente fortemente no modelo de informação da organização ao qual este é aplicado. Apesar disso, baseou-se a geração e manutenção dos perfis de utilizador em processos formais. Deste modo, garante-se uma coerência e generalização que permitem assegurar, quer a correcta utilização da informação, quer a não dependência de nenhum modelo de informação específico.

O servidor UMS referido no estado da arte, necessita de um servidor LDAP para poder funcionar. No caso particular da Universidade de Aveiro, o sistema de informação assenta num sistema LDAP apoiado por um conjunto de serviços web. Contudo, este poderá não ser o caso noutras organizações, em que esta abordagem pode ser seguida (o sistema de informação pode não ser baseado em LDAP, o processo formal poderá ser completamente manual, ...). Assim sendo, e seguindo as recomendações em (Judy et al., 2002; Kobsa & Fink, 2006) o sistema de gestão de perfis é completamente autónomo do sistema de informação onde é aplicado e da tecnologia por ele usada. A nossa abordagem foi a de minimizar os requisitos funcionais e operacionais para o funcionamento do servidor de perfis.

Sempre que um perfil de utilizador é criado ou actualizado, o sistema de gestão de perfis reúne toda a informação requerida num perfil de utilizador. O perfil de utilizador é depois usado para garantir acesso a interfaces específicas no portal. Baseando-se neste, a autoridade de validação do sistema deverá decidir se o utilizador tem ou não acesso a uma determinada operação.

Todos os componentes definidos anteriormente (perfil de utilizador, perfil básico, operação, temas e serviços) são armazenados como ficheiros em formato XML. Em cada um destes componentes base, as referências aos diversos elementos são feitas usando unicamente os seus identificadores. Aquando da sua materialização no perfil de utilizador, estas referências são substituídas pelos elementos completos, aplicados ao perfil. Isto significa que o perfil de utilizador tem toda a informação necessária para que o portal (ou

qualquer outra aplicação que lhe aceda) possa avaliar de forma correcta os privilégios do utilizador. Este processo facilita a fase de recolha de dados sobre o perfil do utilizador, mas torna o processo de gestão mais complexo. O sistema de gestão de perfis foi criado precisamente para ultrapassar esta complexidade.

Usando este gestor, é possível efectuar pesquisas em todo o sistema, facilitando tarefas como o escrutínio e controlo da informação de utilizadores (Judy et al., 2002). A reavaliação de estereótipos (*stereotype reasoning* (Kobsa, Koenemann, & Pohl, 2001)), ou verificação da actualidade das informações contidas no perfil de utilizador, é outro dos aspectos controlados pelo sistema de gestão de perfis. Segundo os autores, *stereotype reasoning* significa avaliar regularmente as condições de activação dos estereótipos existentes e, se estas se verificarem para um dado utilizador, integrar as informações do estereótipo no modelo do utilizador como presunções sobre este. No caso particular dos perfis de utilizador que estamos a definir, as condições de activação serão definidas por intervalos temporais e regras formais, não havendo presunções, mas permissões de acesso a operações.

Funções de gestão, como criar e editar perfis base, definir operações e as suas regras, associar contextos, etc., estão restringidas a utilizadores autorizados pelo sistema de gestão de perfis. Cabe a estes definir quais os utilizadores (ou grupos de utilizadores, mediante os perfis básicos) que podem aceder a determinadas operações, mediante processos formais.

O sistema de gestão de perfis é, na sua natureza, um sistema SOA, visto que encapsula todas as suas funcionalidades e complexidade numa vista abstracta e lógica do seu modelo de funcionamento. A comunicação entre este e as demais aplicações é baseada em serviços web. Assim, para cada um dos componentes foi definido um conjunto de métodos acessíveis via serviços web, que podem ser usados para a sua gestão.

6.1 API - Interface de programação

Todo o sistema de gestão de perfis está acessível via serviços web. Estando criados vários elementos nos quais se subdivide o sistema, e sendo estes semelhantes do ponto de vista funcional (todos necessitam de uma interface CRUD – *create, read, update e delete* (Haim, 1990)), foi implementada uma interface de programação (*Application Programming Interface* – API) genérica que cada gestor de componente em particular posteriormente

integrou, de forma a facilitar a implementação e promover expansibilidades futuras. A Figura 24 mostra o diagrama de classes do sistema de gestão de perfis.

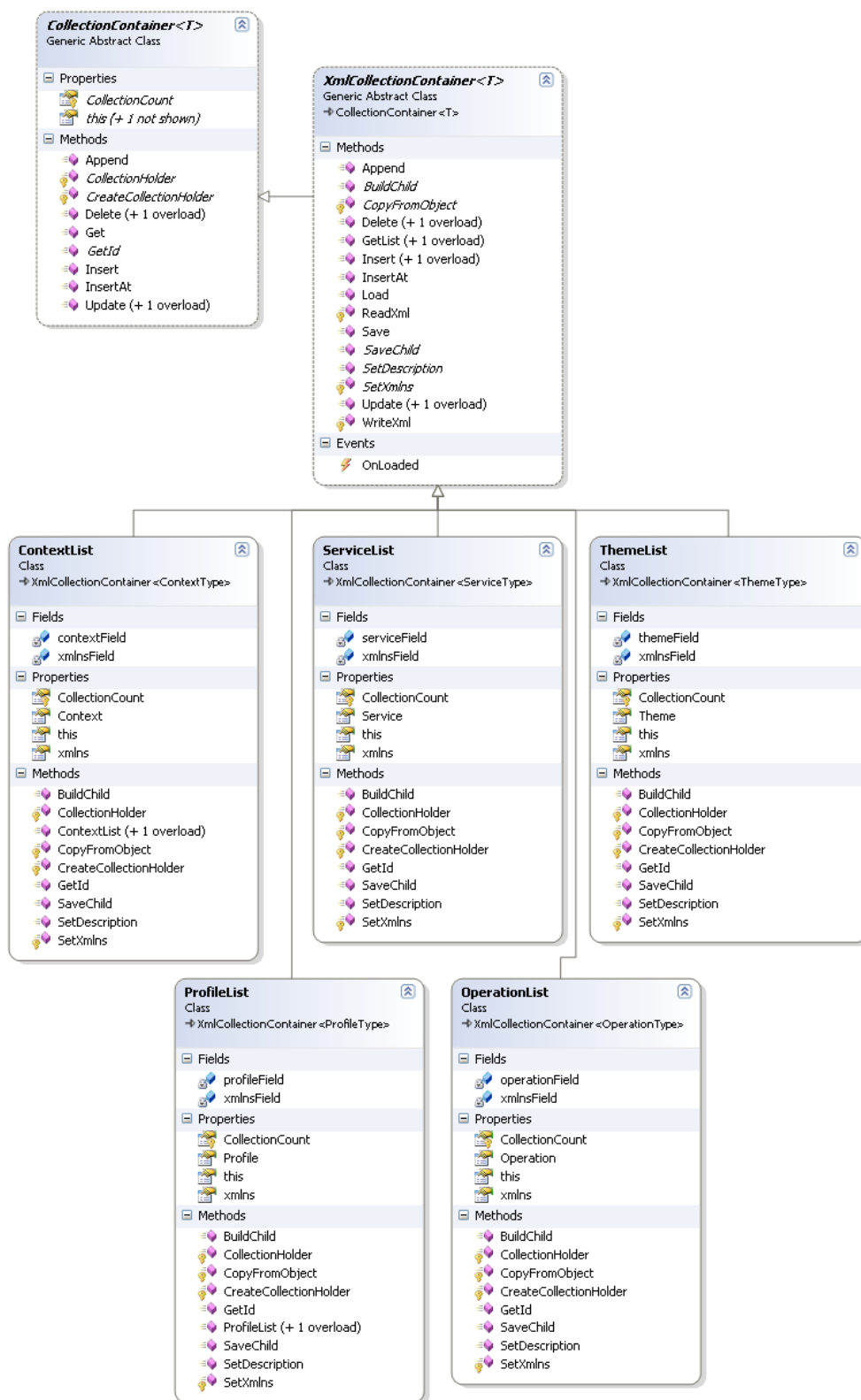


Figura 24 - Diagrama de classes do sistema de gestão de perfis

Como a maioria dos gestores de elementos irá gerir colecções de informação, a classe base para qualquer gestor é genérica e abstracta, com interface para criar, ler, actualizar e apagar elementos de uma colecção, de acordo com o tipo T da mesma (*CollectionContainer<T>*). Os métodos abstractos estão representados em itálico na figura.

Tal como referido anteriormente, a informação será armazenada em ficheiros com formato XML e, seguindo a abordagem da colecção genérica e abstracta, foi definida uma colecção XML genérica e abstracta que providencia as bases para todo o sistema poder trabalhar com documentos XML (*XMLCollectionContainer<T>*).

Alguns dos gestores, como o gestor de serviços (*ServiceList*) e o gestor de temas (*ThemeList*) não requerem atenções específicas ou optimizações em particular, pois o número de temas e serviços – em especial de serviços – são à partida diminutos (dependendo, como é óbvio da complexidade da organização). Contudo, no caso do gestor de operações (*OperationList*), contextos (*ContextList*) e perfis (*ProfileList*) podem ser necessárias optimizações. Como todos os gestores herdam as funcionalidades definidas no *XMLCollectionContainer<T>*, então todos os gestores beneficiam das optimizações que venham a ser feitas.

Cada gestor de elementos é uma classe concreta da classe genérica e abstracta, o que significa que esta apenas terá que se preocupar com as partes específicas (os métodos abstractos definidos nas classes base) para ser considerada eficaz. Os métodos que necessitam de implementação nos gestores são normalmente simples e similares (com pequenas modificações devido a estruturas de informação distintas).

A título de exemplo, considerem-se os métodos que permitem copiar de objecto (*CopyFromObject*) e retornar o identificador (*GetId*) do gestor de serviços (Figura 25) e do gestor de operações (Figura 26).

```
protected override void CopyFromObject(object ob){
    if (ob is ServiceList) {
        ServiceList s1 = (ServiceList)ob;
        this.serviceField = s1.serviceField;
    }
}
public override string GetId(ServiceType item){
    return item.ServiceID;
}
```

Figura 25 - Métodos do gestor de serviços


```
protected override void CopyFromObject(object ob){
    if (ob is OperationList)    {
        OperationList op = (OperationList)ob;
        this.operationField = op.operationField;
    }
}

public override string GetId(OperationType item){
    return item.OperationID;
}
```

Figura 26 - Métodos do gestor de operações

Estes métodos são depois invocados pelas classes abstractas para realizar as tarefas concretas.

6.2 Implicações de segurança

A segurança dos dados dos utilizadores e da organização são tópicos bastante importantes. Se advogamos um sistema de informação único na organização, é necessário estarmos em condições de poder garantir que o utilizador que acede ao sistema, é realmente o utilizador, e não alguém que se tenta passar por ele. Além disso, se o utilizador acede a todos os sistemas com o mesmo par utilizador/palavra passe, será necessário prever o que acontece no caso desses dados serem comprometidos.

No início deste projecto, e devido a esses temores, os serviços da universidade estavam bastante relutantes em fornecer acesso à informação sobre utilizadores. De forma a minimizar o número de possíveis conflitos por questões relacionadas com a segurança, criou-se um plano com duas frentes:

1. Toda a informação acedida aos serviços é apenas para consulta: o portal não pode alterar essa informação;
2. O acesso a certas operações de determinados serviços pode requerer validações de segurança extra. Estes mecanismos de validação estão definidos na Figura 27, que representa a classe nível de segurança (SecurityLevelType), tal como definida no sistema de gestão de perfis.

O nível de segurança pode variar desde o simples par utilizador/palavra passe (nível mais baixo) até uma conjugação de validações de segurança distintas. Esta classe também define alguns dos tipos de segurança tipo:

- BioLevelSecurity – requer sinal biométrico
- CampusLevelSecurity – requer que o utilizador esteja a aceder ao sistema na intranet
- CredentialLevelSecurity – requer um certificado digital
- LowLevelSecurity – par login/password
- PresentialLevelSecurity – requer o acesso a partir de uma localização específica

Com a apresentação desta solução, e com o desenvolvimento do portal, a confiança aumentou de tal forma que futuras operações têm a possibilidade de alterar informações nos sistemas.

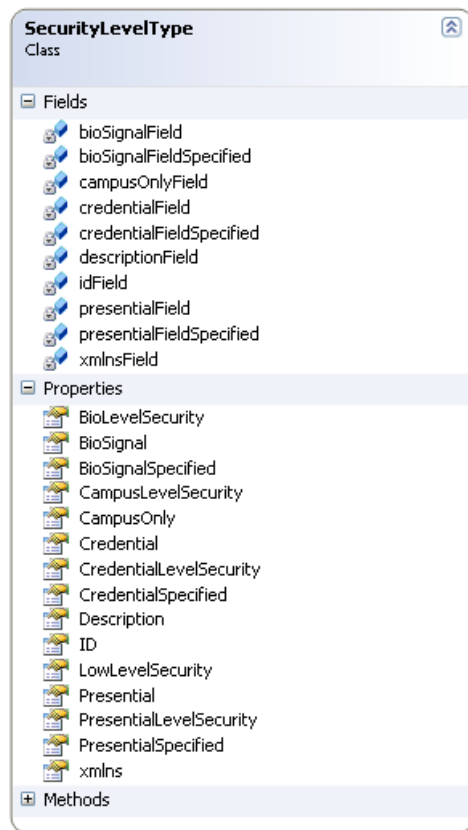


Figura 27 - Classe nível de segurança

6.3 Definição do processo formal

A reavaliação de estereótipos, ou neste caso a actualização do perfil de utilizador, é feita através de processos formais da organização. Uma vez que o sistema de gestão de perfis é independente da arquitectura e sistema de informação da organização, é necessário

providenciar uma forma para especificação de regras que serão posteriormente avaliadas pelo sistema.

Um processo formal na organização, leva a uma mudança na condição do utilizador no sistema de informação desta (independentemente da natureza do sistema). O sistema de informação pode ser baseado em LDAP, base de dados relacional, base de dados XML, ficheiros de texto, ou uma combinação destas (e/ou outras) tecnologias.

A forma mais simples de implementar a interligação entre os processos formais no sistema de informação e o sistema de gestão de perfis, seria colocando um serviço web em funcionamento no sistema de informação da organização, garantindo que este respondia exactamente às necessidades da gestão de utilizadores. Contudo, para ser o menos invasivo possível no sistema anfitrião, o sistema de gestão de perfis deve poder ser expandido para suportar virtualmente qualquer tipo de sistema. Tratando-se de um componente modular, a ligação ao sistema anfitrião poderá ser adaptada de acordo com as características deste. Quando lhe for exigido, este módulo verifica junto do sistema anfitrião, quais os processos formais a que um utilizador se adequa (C. Teixeira et al., 2008c).

Na secção de validação de cada operação, contexto e perfil básico é possível definir regras de acordo com o sistema anfitrião. As regras do processo formal são compostas por uma chave de ligação (*ConnectionString*) que identifica o sistema – ou sub-sistema – onde pesquisar e uma expressão de pesquisa. Exemplos:

- se um dado contexto deve apenas estar disponível para utilizadores no directório LDAP ua.pt, membros do grupo estudante, então a *ConnectionString* deverá ser "LDAP://ua.pt/DC=ua, DC=pt" e a expressão de pesquisa "memberOf=estudante and SAMAccountName={username}";
- se uma dada operação apenas deve estar disponível para utilizadores na base de dados *users* e que se tenham matriculado antes de 2004, então a *ConnectionString* deverá referir a ligação à base de dados e a expressão de pesquisa será "select username from users.student where anoMatricula<2004 and username={username}", assumindo que existe a tabela student onde são registados os estudantes e que esta contém pelo menos dois campos: o nome de utilizador (*username*) e o ano de matriculo (*anoMatricula*).

A expressão {username} é uma representação interna do utilizador e é substituída aquando da verificação da condição. Neste momento, este sistema suporta pesquisas simples em LDAP e bases de dados relacionais.

Este processo é complementar ao período temporal de validação definido também em todos os elementos. O período temporal deve ser usado para períodos bem definidos, ou no pior caso, com uma duração comum para acesso a uma dada operação. As regras dos processos formais reforçam as restrições temporais e podem revogar ou alterar as validações temporais.

6.4 Acesso por diferentes aplicações

Tendo em consideração a linha de pesquisa (perfis dinâmicos em organizações), o sistema de gestão de perfis pode interagir com diferentes aplicações, todas acedendo aos mesmos perfis de utilizador. Neste cenário, os contextos poderão ser de uma importância acrescida, visto que o mesmo contexto poderá conter operações para diferentes aplicações, alargando a definição de contexto do nosso caso de estudo – operações que fazem sentido juntas numa página web – para operações que simplesmente se relacionam, independentemente do meio de apresentação das mesmas.

A Figura 28 mostra um diagrama de camadas com a interacção entre as aplicações, o sistema de gestão de perfis e o sistema anfitrião.

Usando como referência um modelo de três camadas, apresentamos uma interface baseada nas próprias aplicações, uma camada lógica baseada no sistema de gestão de perfis, perfis de utilizadores, operações e validação central, e uma camada de dados onde está representado o sistema de informação anfitrião.

Na camada de interface, entre outras possíveis aplicações está representado o caso de estudo my.ua. Todas as aplicações autenticam os utilizadores por intermédio da validação central, acedem aos perfis, autorizam utilizadores mediante o sistema de gestão de perfis e autorizam o acesso a operações concretas, baseando-se nas definições de segurança definidas na operação.

Na Figura 28 estão exemplificadas várias operações:

- A operação A não exige nenhum grau de segurança acrescido,
- A operação B requer informação biométrica, como a leitura de uma impressão digital,
- A operação C requer um certificado digital e
- A operação D é apenas autorizada para utilizadores no campus.

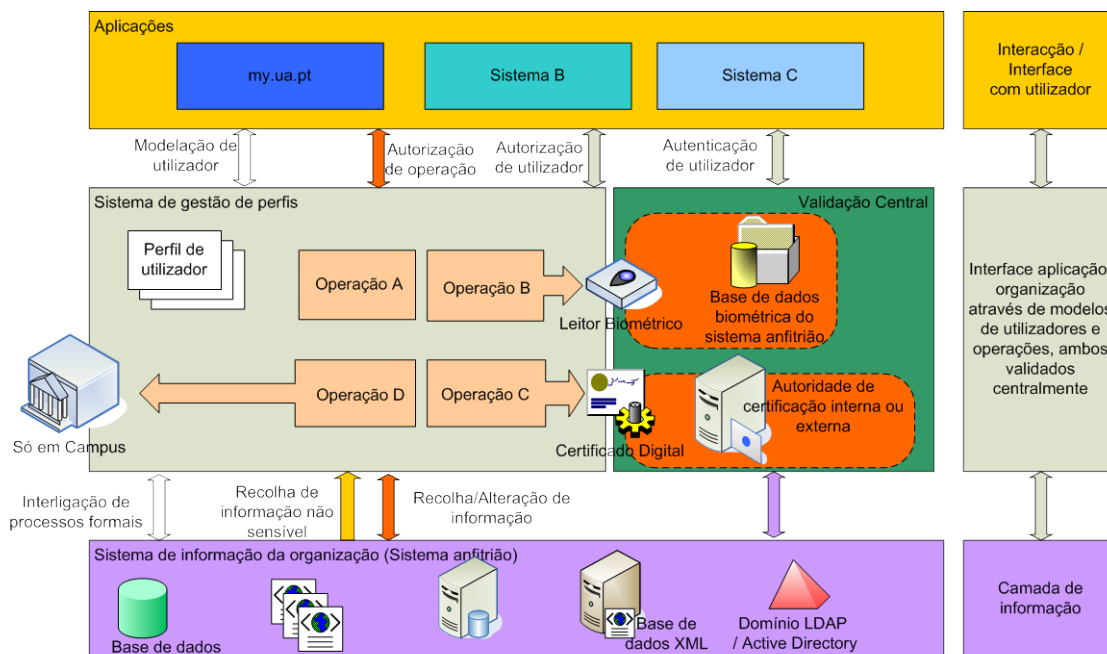


Figura 28 - Modelo de três camadas aplicado à interação entre aplicações, sistema de gestão de perfis e sistema anfitrião.

A interação entre a camada de informação e a camada lógica é feita mediante a interligação dos processos formais e mediante as operações. À camada de interface, é possível aceder, actualizar, inserir e remover informação sensível e não sensível. Como melhor prática de segurança, operações que insiram, modifiquem ou apaguem informação (e/ou que acedam a informação sensível), apenas serão autorizadas após uma segunda validação de segurança.

Como expresso na Figura 28 as camadas são completamente independentes. Deste modo, cada camada poderá evoluir, de acordo com as suas necessidades, sem interferir com os demais sistemas.

6.5 Comparação com servidores de modelação de utilizadores

6.5.1 Semelhanças

Qualquer sistema de modelação de utilizadores deverá estar de acordo com as seis características (ou requisitos) definidos por Alfred Kobsa e Josef Fink:

“(...) a disponibilidade de esquemas pré-definidos que sejam relevantes para a modelação de utilizadores e que possam ser expandidos livremente; a [capacidade de] gestão de informação distribuída, replicação, performance e escalabilidade, adesão a formatos abertos, e a gestão consistente (...)” adaptado de (Kobsa & Fink, 2006)

No nosso caso:

- Os esquemas pré-definidos são os perfis básicos.
- A informação respeitante aos modelos de utilizador é armazenada de forma independente e pode ser gerida pelo sistema de ficheiros.
- Considerando que a informação é armazenada directamente no sistema de ficheiros, podem ser empregues técnicas de replicação para assegurar a redundância e consistência da informação.
- O sistema tem um desempenho rápido em operações de recolha de informação – a tarefa mais importante e mais comum em sistemas com taxas de actualizações diminutas comparativamente às taxas de leitura.
- Como todo o sistema é baseado em serviços web e SOAP, a escalabilidade dos pontos de acesso é simples de se conseguir, recorrendo a aglomerados de servidores web (*web farms*), e a escalabilidade do repositório de informação pode ser conseguida através de sistemas dedicados para armazenamento de informação em rede.
- A API do sistema de gestão de perfis é SOAP, facilitando a interoperabilidade e,
- Os processos formais, conjuntamente com as restrições de segurança e validação em cada elemento facilitam a gestão de consistência e coerência de informação.

6.5.2 Diferenças e melhorias

A API de acesso é baseada em SOAP. Contrariamente aos sistemas de modelação de utilizador apresentados, o sistema de gestão de perfis é independente das aplicações e do sistema anfitrião.

A informação é armazenada em ficheiros XML. Os perfis básicos são estruturados e definidos de forma hierárquica, culminando em perfis de utilizador também estruturados. Todos os elementos têm uma definição estruturada em três partes: identificação, validação e interface.

A actualização do perfil de utilizador é feita não só pelo utilizador e aplicações, mas também pelo próprio sistema anfitrião, através dos processos formais e das regras definidas nestes.

De um ponto de vista conceptual, há algumas diferenças que convém esclarecer entre o modelo proposto e os modelos definidos no estado de arte. Estas são indicadas na Tabela 5.

Ao contrário do modelo de utilizador definido no estado de arte, o perfil de utilizador pode nalgumas circunstâncias, ser semelhante (ou mesmo idêntico) para vários utilizadores. Dois trabalhadores do mesmo gabinete, ou dois estudantes matriculados no mesmo curso podem ser exemplos dessa situação. Contudo, acreditamos que esta não é uma fraqueza do sistema, visto que o principal objectivo é entregar a informação certa aos utilizadores.

Mesmo no caso de dois perfis serem baseados nos mesmos perfis básicos, a informação pessoal contida em cada serviço poderá levar a que estes sejam vistos como únicos. Ainda assim, o modelo proposto permite a criação de perfis básicos bastante específicos, maximizando a granularidade do sistema, de acordo com as necessidades do modelo de negócio da organização.

	Estado de arte de modelação de utilizadores	Modelo proposto
Estrutura dos conjuntos de características	Maioritariamente não estruturado	Hereditário
Principais características em cada conjunto	<ul style="list-style-type: none"> • Necessidades • Objectivos • Conhecimento e • Interesses 	Modelos transaccionais; regras de segurança e propostas de arranjo de interface definidos pelos diversos elementos do perfil: <ul style="list-style-type: none"> • O que o utilizadores pode fazer • Onde a informação é vista
Criação e verificação da actualidade de perfis	Baseado em condições de mudança (<i>triggers</i>)	Baseado em processos formais
Descrição das características de utilizador	Parte da previsão do perfil	Perfis básicos
Actualização de perfis	<ul style="list-style-type: none"> • Normalmente apenas é considerado a actualização do perfil de utilizador; • Actualizações de estereótipos não estão necessariamente relacionadas com actualizações de perfis de utilizadores. 	<ul style="list-style-type: none"> • Actualização directa: actualiza um perfil de utilizador específico; • Actualização indirecta: a actualização de perfis básicos é reflectida nos perfis de utilizador.

Tabela 5- Principais diferenças e melhorias entre o sistema de gestão de perfis e o estado de arte

6.5.3 Vantagens e limitações

Baseando-se numa API disponibilizada sob a forma de serviços web, o sistema de gestão de perfis é independente do sistema anfitrião. O acesso aos serviços web está limitado a utilizadores e serviços/aplicações autorizados. Este acesso pode ser definido por cada método disponibilizado.

Este sistema pode ser usado por qualquer aplicação da organização, uma vez que delega nesse sistema anfitrião as alterações a dados do utilizador. Este sistema não tem como objectivo ser completamente adaptável, mas sim ser um ambiente personalizado por utilizadores e serviços de uma organização, baseando-se em perfis e responsabilidades dinâmicas dos utilizadores na organização.

Ao confiar em serviços web para gerir todo o processo, há vários aspectos que devem ser considerados:

1. A comunicação por meio de serviços web induz um acréscimo de tráfego na rede. O acréscimo de tráfego induzido pelo protocolo SOAP pode ser prejudicial, e cabe à equipa de desenvolvimento avaliar os prós e contras da adopção desta tecnologia. Da experiência obtida, o tráfego adicional é um pequeno preço a pagar por interfaces standard, quando o que está em causa é a capacidade de inter-operar com outros sistemas.
2. A comunicação na rede tem que ser segura. Para a implementação dos serviços web deverão ser usados canais seguros (HTTPS em detrimento de HTTP). Esta exigência acarreta um novo aumento de tráfego necessário para a comunicação.
3. Os serviços web podem potencialmente ser usados por qualquer utilizador. Como referido anteriormente, o acesso aos métodos disponibilizados pelo sistema de gestão de perfis está limitado aos utilizadores constantes em listas de controlo de acessos, pelo que este problema está minorado.

6.6 Conclusão

O sistema de gestão de perfis proposto e o modelo de perfis de utilizador, têm como finalidade não a construção de interfaces adaptáveis, mas o acesso personalizado a serviços em ambientes corporativos, baseado em perfis de utilizadores dinâmicos. Para tal, propomos um modelo de perfis de utilizador hereditário, definido por elementos de perfil com modelos transaccionais, regras de segurança e propostas de arranjo gráfico de operações, onde a verificação da actualidade do perfil é baseada em processos formais. Os perfis de utilizador podem ser actualizados de forma directa e indirecta e as permissões atribuídas definidas em modelos denominados de perfis básicos. Estes perfis de utilizador são geridos pelo sistema de gestão de perfis, cujo modelo de funcionamento é distinto dos sistemas de modelação de utilizador mencionados no estado de arte.

O sistema de gestão de perfis é independente do sistema anfitrião, e das aplicações que o usam, podendo ser usado para extrair, inserir, actualizar e remover características de utilizadores, através de um utilizador ou grupo em particular. Este acesso e/ou alteração pode ser solicitada pelo utilizador, por uma aplicação ou pelo sistema anfitrião. Sendo um sistema independente, este é visto como uma caixa negra, com interfaces conhecidas para

acesso aos seus métodos, permitindo a evolução natural, quer do sistema de gestão de perfis, quer das aplicações que o usam, sem que isso comprometa toda a arquitectura.

Para este sistema propomos uma API baseada em SOAP, com controlo de acesso a cada método constante na API, por parte de aplicações e utilizadores. Contudo, o sistema não pode existir sozinho, visto que requer um sistema anfitrião (seja este qual for) que defina as regras de criação e actualização de perfis de utilizador. Tendo em consideração que o sistema de gestão de perfis foi construído para uso em organizações, consideramos que esta característica não pode ser vista como um defeito do sistema.

Apesar de neste momento o sistema estar já em produção, há ainda algumas melhorias que poderão ser feitas. As regras de processos formais, uma das inovações, devem poder ser usadas em diversos repositórios de dados de forma transparente, com pesquisas complexas que interliguem vários repositórios. De momento, apenas pesquisas simples em sistemas de bases de dados e LDAP estão disponíveis.

7 my.ua.pt – O sistema

O sistema desenvolvido é composto por duas partes: a plataforma my.ua e os componentes a visualizar. Se, por um lado, a plataforma é genérica e adaptável a vários cenários de utilização, as operações e contextos são específicos do cenário onde o portal será colocado em produção. Deste modo, importa antes de mais, analisar quais as operações e contextos (mínimos) necessários para o my.ua.

Atendendo às especificidades dos utilizadores (e de acordo com a lógica de perfis definida anteriormente), foram concebidos cinco contextos de interação:

- Entrada
- Correio
- Aulas
- Repositório
- Favoritos

A Figura 29 apresenta esquematicamente os contextos e operações definidas para o my.ua. Estes são os contextos e operações comuns, mas o propósito final será permitir que os serviços desenvolvam e publiquem os seus próprios componentes no sistema.



Figura 29 - Esquemática dos contextos e componentes (operações) desenvolvidas

Para cada um destes contextos, serão criados componentes de operação (como definido no perfil de utilizador) que depois de associados a um ou mais contextos, estarão disponíveis de acordo com o perfil do utilizador.

No contexto de **entrada**, e por se tratar da página de entrada, depois da identificação do utilizador, este tem como propósito principal mostrar-lhe um resumo da informação que lhe diga respeito. Na sua maioria, os componentes desta página são apenas de leitura. Analisando a informação disponível nos diversos sistemas, foram criados os seguintes componentes:

- Leitor de notícias
- Informação pessoal
- Agenda
- Aulas de hoje (a leccionar ou a assistir)
- Notificações
- Sumário de mensagens de correio electrónico.

O contexto de **aulas** é o contexto onde deverá estar concentrada parte da interacção com o utilizador, sobre disciplinas e correspondentes aulas. De acordo com este propósito, foi criado um componente que permite ao utilizador navegar na sua informação, desde o seu horário semanal, até à informação sobre os sumários e colegas/alunos de uma determinada turma. Este contexto é apenas disponibilizado a utilizadores com perfil de aluno ou de docente.

O contexto **correio** contém o componente de correio electrónico e o componente de agenda. Trata-se de componentes agregadores e portanto com suporte a várias contas de utilizador.

Para o contexto de **favoritos** foi desenvolvido um componente de leitor de conteúdos noticiosos recebidos por RSS.

O contexto **arca** visa fornecer mecanismos para acesso a espaço de armazenamento disponibilizado centralmente pelo Centro de Informática e Comunicações. Além de espaço de armazenamento, é também possível a qualquer utilizador manter a sua página pessoal. Para interacção com estes serviços, foram desenvolvidos dois componentes:

- Explorador de ficheiros
- Editor de páginas HTML.

O contexto de **definições** visa permitir a configuração e uma maior personalização do sistema por parte do utilizador. No fundo, existe um componente de definições por cada componente, referido anteriormente, que seja passível de ser personalizado.

7.1 Leitor de notícias

O que se pretende com este componente é disponibilizar aos utilizadores, informação e conteúdos relacionados com a Universidade e adequados ao perfil do utilizador.

Este componente recebe as notícias disponibilizadas por outros sistemas, por intermédio de documentos estruturados de acordo com *RDF Site Summary* (RSS (Begeed-Dov et al., 2001)). Os RSS consumidos são fornecidos pelo serviço noticioso da Universidade (uaOnline). De acordo com os temas associados ao perfil, o uaOnline entrega um feed representativo das notícias que poderão ter interesse para o utilizador. Este componente é disponibilizado a todos os perfis.

Neste componente é possível definir o número de notícias a apresentar. Por omissão são mostradas quatro notícias.

A Figura 30 mostra as notícias disponibilizadas na página de entrada de um determinado utilizador.

noticias de interesse

Mobilidade sustentável e gestão de recursos hídricos na antena do «Click»



Murtosa e Oliveira de Frades são vilas exemplares em matéria de mobilidade sustentável. No âmbito de um projecto da Agência Nacional para o Ambiente, os investigadores da Universidade de Aveiro definiram estratégias de desenvolvimento, que se traduzem em novas oportunidades para estes municípios, respeitando o ambiente e oferecendo qualidade de vida a quem lá mora. Conheça o projecto, em pormenor, na emissão deste sábado do «Click», o programa da UA que é emitido às 13h30, na Antena 1.

DeCA participa na audição integral da obra organística de Olivier Messiaen



Decorre, de 29 de Novembro a 15 de Dezembro, na Sé Patriarcal de Lisboa, a audição integral da obra organística de Olivier Messiaen, no âmbito das comemorações do centenário do seu nascimento. O concerto deste Sábado, 6 de Dezembro, às 21h30, está a cargo de Edite Rocha e dos alunos da UA Vasco Soeiro e Ricardo Toste, que apresentam Meditations sur le Mystère de la Sainte Trinité (1969).

Este mês há uma nova história contada «Domingo de manhã na barriga do caracol»



As histórias com enredos de ciência que se contam aos 1º e 3º Domingos do mês, na Fábrica, conhecem um novo capítulo. «Obrigado, Camião Azul» é a próxima narrativa que as crianças, entre os três e os oito anos de idade, podem conhecer nos dias 7 e 21 de Dezembro, às 11h00, na «barriga do caracol». As inscrições já estão abertas.

Figura 30 - Aspecto do leitor de notícias integrado

7.2 Informação pessoal dos serviços académicos

Este componente tem por objectivo mostrar informação pessoal do utilizador, indicando-lhe o seu vínculo associativo com a Universidade.

Este componente é apenas disponibilizado a utilizadores que tenham como perfil base o perfil “Aluno”.

A Figura 31 mostra informação que apesar de ser do conhecimento do utilizador, poderá ser alvo de actualização.

informação



dáudio teixeira
9115 - doutoramento em
engenharia informática

Figura 31 - Informação PACO

Importa referir que, o importante neste componente, é a demonstração da potencialidade de se ter informação e dados específicos para um determinado conjunto de utilizadores. Neste caso em concreto, esta informação é apenas para utilizadores que sejam alunos.

7.3 Aulas de hoje

Este componente tem como objectivo fornecer a lista de aulas a assistir ou a leccionar no dia.

Esta informação é consultada através de um web service disponibilizado pelos serviços académicos. Atendendo a que se trata de informação que sofre poucas ou nenhuma alterações durante um semestre, foram pensados mecanismos de armazenamento desta informação no perfil my.ua do utilizador, de modo a minimizar o número de interacções com o serviço visado, garantindo desta forma um aumento de disponibilidade do mesmo para resposta a outros tipos de pedidos. A actualização desta informação é despoletada de forma activa pelo utilizador ou de forma passiva, após um determinado período temporal previamente configurado. Este componente está disponível a utilizadores com perfil base “Aluno” ou “Docente”.

É possível especificar o tempo entre actualizações. Por omissão este valor é de sete dias.

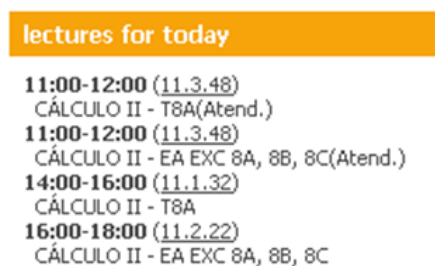


Figura 32 - Informação de aulas para o dia

A Figura 32 representa uma listagem de aulas a leccionar por um determinado docente. No caso de o utilizador estar associado a um perfil de aluno, então terá a lista de aulas a assistir.

7.4 Cliente de correio electrónico e agenda

Este componente tem como objectivo integrar no portal o serviço de correio electrónico e agenda fornecido pela Universidade, permitindo o acesso por parte do utilizador a todas as suas contas de correio electrónico. O utilizador poderá também integrar contas de correio e agenda externas à Universidade. Dado que cada utilizador pode ter várias contas de correio electrónico, o componente desenvolvido permite configurar todas as contas às quais o utilizador pode aceder usando o protocolo *Internet Message Access Protocol 4* (IMAP4).

Assim, estamos perante um cliente agregador de correio electrónico, que permite numa mesma interface a consulta e envio de mensagens de diversas contas, muito à semelhança de qualquer cliente de correio instalado num computador pessoal, como é o caso do Outlook da Microsoft, ou do Thunderbird da Mozilla, onde a principal distinção é o facto de ser um cliente na rede.

Um dos componentes identificados como fulcral para a adopção do sistema pela comunidade foi o componente de correio electrónico. Com este, passa a existir uma alternativa ao cliente web disponibilizado pelo servidor da universidade. Contudo, uma vez que este é alojado num serviço que não o próprio servidor, significa que a sua lógica de funcionamento será dupla, pois se para o servidor de correio este não é mais que um novo cliente de correio, para o utilizador este serviço será visto como o seu serviço de correio electrónico. Esta dupla relação acarreta problemas a diversos níveis, sendo que os problemas de performance e os de confidencialidade são os mais importantes a ter em consideração.

A principal causa para os **problemas de performance** é, como referido, este componente estar separado do servidor, a fazer um papel que normalmente está reservado a uma interface que lhe é mais próxima. Assim, é necessário comunicar por intermédio dos protocolos standard para troca de mensagens de correio electrónico entre servidor e cliente. Deste modo, e apesar da norma escolhida (IMAP4) permitir o acesso rápido a muitas das funcionalidades do sistema, é de referir que algumas operações são forçosamente mais lentas do que num cenário de aplicação cliente feita à medida para um determinado servidor. De entre as funcionalidades em questão é de mencionar a construção de agenda de cliente: neste caso é necessário aceder a todas as mensagens que armazenam informação de agenda e compilar, em tempo real, essa informação, de modo a gerar a agenda do utilizador ligado por intermédio deste cliente. A alternativa seria armazenar localmente a informação para um mais rápido acesso, o que traria problemas acrescidos de escalabilidade. Esta questão poderá trazer problemas de performance, se não for considerada na altura de desenvolvimento.

Este componente acede à conta de correio electrónico usando as credenciais do utilizador autenticado, o que pode acarretar **problemas de confidencialidade**. É necessário dar garantias de que estes dados são usados *apenas* neste acesso e que, mais importante, não há

possibilidade de aceder indevidamente a estes dados. Neste caso, o par login/password do utilizador são guardados temporariamente no servidor – durante o período em que o utilizador está ligado no sistema – e apenas usados para o fim em causa. Foram esboçadas soluções que permitiriam minimizar o impacto desta questão usando um protocolo específico do servidor de correio em uso pela Universidade. No entanto, por imposições temporais de entrada em produção, tais propostas não passaram à fase de implementação.

Este componente é um cliente de correio electrónico completo, cujas principais funcionalidades são:

- Envio, resposta e reencaminhamento de mensagens;
- Gestão das mensagens e caixas de correio (opções de mover, copiar e apagar);
- Visualização de qualquer pasta na forma de agenda, perscrutando as mensagens no seu interior em busca de informação de marcações na agenda (Figura 34);
- Interpretação automática do conteúdo de mensagens recebidas, para verificação do tipo de mensagem – normal ou pedido de marcação de agenda;
- Marcação e envio de pedidos de agendamento de tarefas para diversos destinatários;
- Gestão dos diversos parâmetros associados a uma conta de correio electrónico acedido via IMAP.

Além das funcionalidades descritas, importa referir que, no caso dos utilizadores associados à Universidade, e uma vez que todos têm uma conta de correio electrónico associada, no primeiro acesso de um utilizador ao sistema, este componente encarrega-se de configurar de forma automática essa conta. Caso o utilizador pretenda, poderá configurar as contas extra que entender, sendo que nesse caso os dados de acesso à mesma serão armazenados no perfil do utilizador de forma cifrada. São exemplo de utilizadores com múltiplas contas, os detentores de cargos de gestão, pois para além da sua conta pessoal, possuem também uma conta associada ao cargo.

Este componente permite uma vista segmentada do correio electrónico e agendas de um utilizador, de acordo com o número de contas configuradas. A agregação de informação é efectuada nos componentes de resumo associados tipicamente à página de entrada de um utilizador.

A Figura 33 apresenta o aspecto geral do cliente de correio electrónico, estando visíveis a lista de pastas, lista de correio e a visualização de uma mensagem.



Figura 33 - Aspecto geral do cliente de correio electrónico desenvolvido

A Figura 34 apresenta a interface de agenda do componente. Nesta é perceptível a existência de três áreas distintas: a legenda de cores (à esquerda), o calendário detalhado (ao centro) e a vista de calendário mensal sumariado (à direita). As cores dos eventos marcados são atribuídas em função da categoria definida pelo utilizador para o evento. A vista central permite uma visualização alternativa de um dia, de 5 dias (segunda a sexta-feira) e de 7 dias (semana completa).

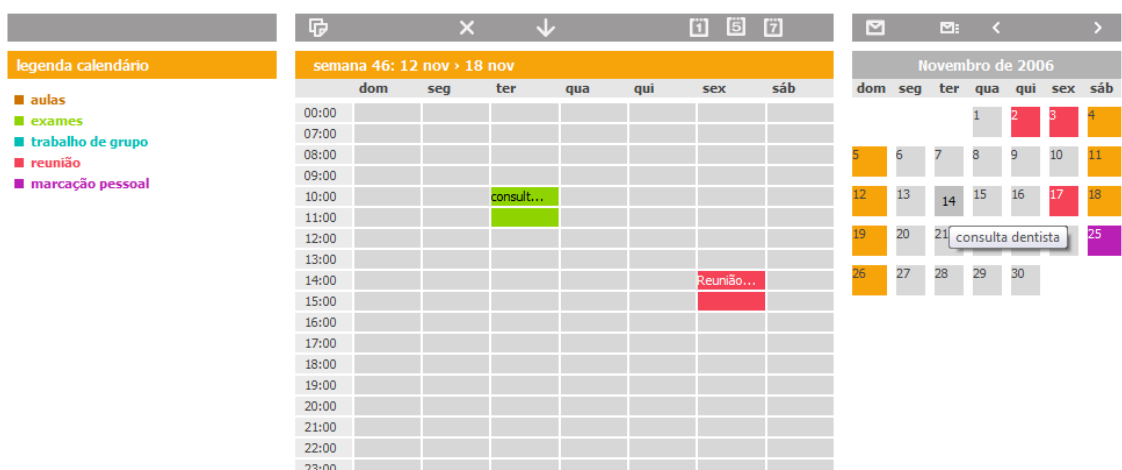


Figura 34 - Interface agenda do componente de correio electrónico

7.5 Mensagens gestão documental

O objectivo deste componente é permitir que o utilizador verifique o estado dos vários processos que dele dependem, ou de que é dependente.

Este componente é disponibilizado a funcionários de secretaria e pivôs departamentais e outros funcionários que, pelas suas características, lidem com requerimentos e formulários, ou que necessitem de estar ao corrente do estado de determinadas operações relacionadas com os mesmos. Trata-se de um serviço especializado, disponível apenas a utilizadores com um determinado perfil, que lhes fornece mensagens sobre o sistema de gestão documental da UA. O acesso à informação é feito mediante um web service disponibilizado pelo serviço de gestão documental da Universidade de Aveiro.

A Figura 35 mostra uma listagem de vários pedidos efectuados ao utilizador que acede ao my.ua, ordenados cronologicamente. No caso particular deste utilizador, não há mensagens a reportar atrasos, caso contrário, seriam também mostradas, com o destaque devido.

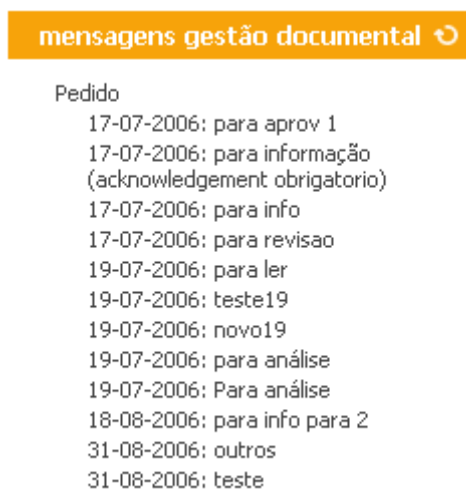


Figura 35 – Componente informativo de mensagens de workflow na gestão documental

7.6 Sumário de mensagens de contas de correio

Este componente tem como objectivo disponibilizar, de forma sucinta, a lista de novas mensagens que um utilizador tenha nas suas diversas contas de correio electrónico.

Este componente extrai informação sobre quais as contas do utilizador através do seu perfil my.ua, que é actualizado por intermédio do componente de correio electrónico referido anteriormente.

A Figura 36 mostra o componente com uma configuração de três contas de correio electrónico, a lista de caixas em cada uma delas e o número de mensagens não lidas por cada caixa.

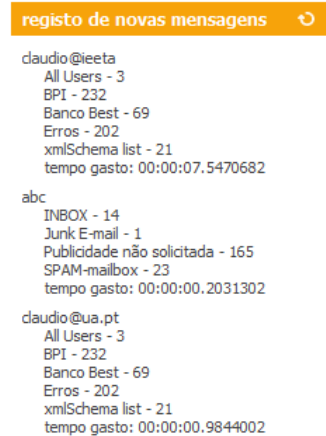


Figura 36 - Sumário de novas mensagens de correio electrónico

7.7 Agenda pessoal

O objectivo deste componente é disponibilizar, numa interface agregadora, eventos de interesse e marcações pessoais de um utilizador num determinado período temporal. Este componente não pode ser confundido com a agenda proporcionada pelo cliente de correio electrónico. Este componente agrega a informação recolhida pelo cliente de correio e agenda com informação proveniente de outras fontes.

Mais do que uma simples agenda, trata-se de um agregador de informação temporal compilada dinamicamente a partir de diversas fontes:

- Serviços académicos – informação de aulas a leccionar ou assistir
- Agenda universitária – serviço noticioso associado ao uaOnline que disponibiliza informação sobre acontecimentos dos mais variados temas, dentro e fora da universidade
- Calendários pessoais – de cada conta de correio electrónico previamente configurada são compilados os apontamentos pessoais aí marcados.

A Figura 37 ilustra a agenda semanal de um utilizador. Cada barra corresponde a um evento, seja um acontecimento ou uma marcação, e cada cor corresponde a uma categoria (pessoal, reunião, aulas, desporto, etc.).

Para saber mais detalhes sobre um dos eventos, basta “passar o rato” sobre o evento em questão. Na Figura 37 está detalhado o lançamento do livro de Carlos Fiolhais.



Figura 37 - Agenda semanal (vista 1)

Sendo um componente agregador da informação considerada relevante pelo utilizador, importa permitir-lhe um grau de configuração e personalização que maximizem o seu agrado com o sistema. Deste modo, o utilizador tem a possibilidade de escolher o que deseja que esteja presente na agenda. A Figura 38 mostra a mesma agenda semanal de um utilizador que optou por remover todas as notificações proporcionadas pelos feeds temáticos do uaOnline e pela sua agenda pessoal, concentrando-se apenas no seu horário lectivo.

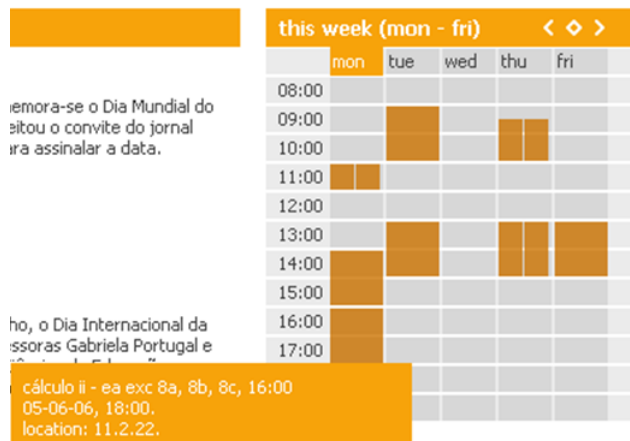


Figura 38 - Agenda semanal (vista 2)

Finalmente, a Figura 39 mostra a agenda semanal de um utilizador que optou por remover todas as notificações à excepção dos seus agendamentos pessoais (constantes nas suas contas de correio electrónico).

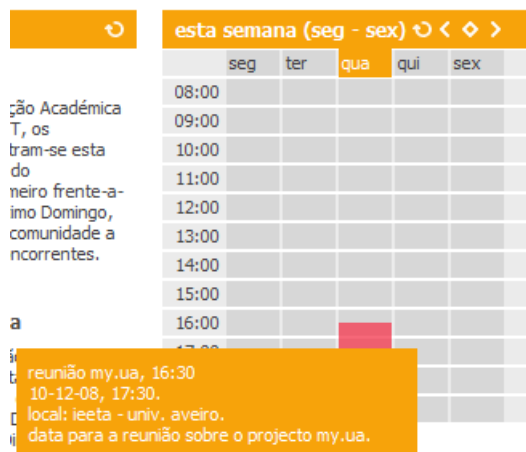


Figura 39 - Agenda semanal (vista 3)

Na Figura 40 podemos observar a interface de configuração da agenda pessoal que é disponibilizada aos utilizadores. Nesta é possível alterar dados como o número de dias a mostrar (1, 5 ou 7), o intervalo de tempo (30 ou 60 minutos) e as categorias de eventos seleccionadas (com distinção entre as pessoais e as de campus).

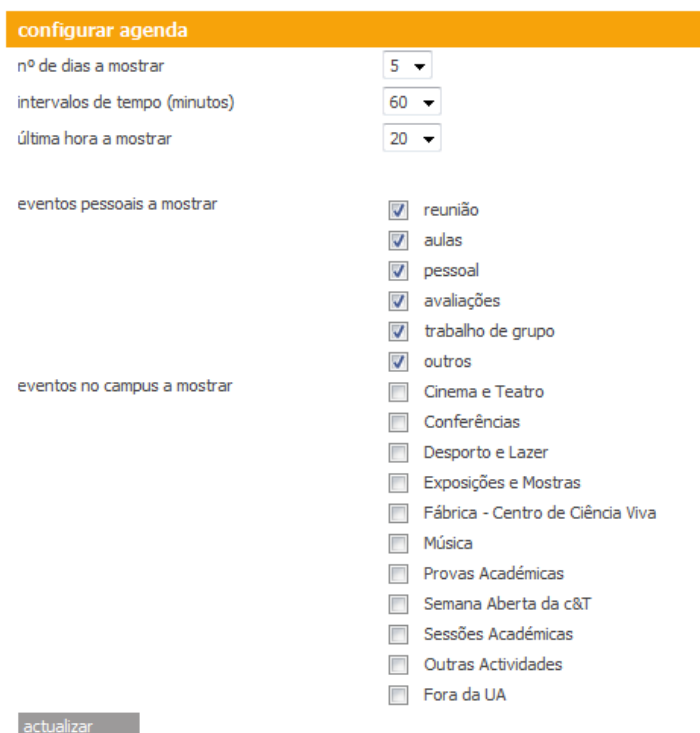


Figura 40 - Interface de configuração de agenda

7.8 Informação dos serviços académicos

No caso de o utilizador ter o perfil de aluno ou docente, terá acesso a uma página específica que lhe servirá de suporte ao seu dia-a-dia lectivo. Trata-se de um componente com várias funcionalidades integradas: horário lectivo, informação de disciplinas e informação de turmas.

7.8.1 Horário lectivo

O objectivo deste componente é mostrar o horário lectivo de um aluno ou docente.

A informação mostrada neste componente é a mesma que a mostrada no componente “Aulas de hoje”, mas com uma visualização mais detalhada.

A Figura 41 apresenta a informação do horário de aulas de um determinado utilizador. Podemos observar que, para além do nome da disciplina e do horário da mesma, há de forma imediata informação sobre o local onde será leccionada.

categorias		horário aulas				
horário		seg	ter	qua	qui	sex
↑ 1º semestre						
arquitectura de computadores i	tp3				arquite...	
	p4a				104/106/119	
análise matemática i	tpr4		análise...	análise...		análise...
		álgebra...	11.1.32	11.2.7		12.2.11
análise numérica	tp4		arquite...		arquite...	
	p4a		anf.iv		anf.iv	
álgebra linear	tpr2					
↓ sem informação de semestre						
				álgebra...		álgebra...
				11.1.27		23.3.10
						análise...
			análise...		análise...	11.1.32
			28.2.20		12.2.14	

análise matemática i - tpr4, 17:00
 02-12-08, 19:00.
 local: 28.2.20.

Figura 41 - Horário lectivo

7.8.2 Informação de disciplinas

O objectivo deste componente é permitir a consulta de informação sobre uma determinada disciplina.

A informação apresentada sobre a disciplina é recolhida dinamicamente por intermédio de web-services. A informação passível de ser consultada sobre a disciplina é a informação geral do funcionamento da mesma, e que está acessível publicamente: área científica, docente responsável, objectivos, conteúdos, etc. A grande diferença é que neste caso é um

discente ou docente da mesma que acede à informação, pelo que não necessita de procurar num local público a informação que pretende. Além disso, nesse sistema público tem acesso a dados respeitantes a disciplinas que não lhe dizem respeito, o que poderá não ser do seu interesse.

A Figura 42 mostra a informação da disciplina de Arquitectura de Computadores I. Como se vê no painel central, há informação diversa, conteúdos, objectivos, etc. No painel da esquerda, sob o nome da disciplina temos as diversas turmas a que o utilizador está associado. Neste caso particular, o utilizador está associado às turmas TP3 e P4a. As hiperligações subjacentes a TP3 e P4a permitem o acesso a informação própria da turma.

The screenshot shows a web interface for a course. On the left, there is a sidebar with a 'categorias' header and a list of subjects: '1º semestre', 'arquitectura de computadores i', 'tp3', 'p4a', 'análise matemática i', 'tpr4', 'análise numérica', 'tp4', 'p4a', and 'álgebra linear', 'tpr2'. Below this is a 'sem informação de semestre' section. The main content area has a header 'arquitectura de computadores i > 2226'. It lists course details: 'área científica: Informática / Arquitectura dos Sistemas Computacionais', 'créditos: 6', 'escolaridade (t|tp|p): 0|3|2', and 'responsável: José Luis Costa Pinto de Azevedo'. It also includes sections for 'objectivo(s)', 'conteúdo(s)', and 'requisitos'.

categorias	arquitectura de computadores i > 2226
<p>horário</p> <p>↓ 1º semestre</p> <p>arquitectura de computadores i</p> <p>tp3</p> <p>p4a</p> <p>análise matemática i</p> <p>tpr4</p> <p>análise numérica</p> <p>tp4</p> <p>p4a</p> <p>álgebra linear</p> <p>tpr2</p> <p>↓ sem informação de semestre</p>	<p>área científica: Informática / Arquitectura dos Sistemas Computacionais</p> <p>créditos: 6</p> <p>escolaridade (t tp p): 0 3 2</p> <p>responsável: José Luis Costa Pinto de Azevedo</p> <p>objectivo(s)</p> <ul style="list-style-type: none"> Compreender a organização dos computadores digitais. Adquirir familiaridade com a arquitectura de microprocessadores através da programação em assembly. Compreender a estrutura interna dos processadores. Conhecer as formas de representação da informação nos computadores digitais, com relevo para a de representação da informação numérica (inteiros e vírgula flutuante) e as operações aritméticas básicas. <p>conteúdo(s)</p> <ol style="list-style-type: none"> Organização dos computadores digitais numa perspectiva funcional, através da descrição do repertório de instruções e da programação em assembly. Tradução em assembly dos mecanismos das linguagens de alto nível. O processo de Assemblagem. Aritmética de vírgula fixa e de vírgula flutuante. Estrutura interna básica do processador. Etapas de processamento das instruções. Introdução às arquitecturas de processadores com pipeline. <p>requisitos</p> <p>Os alunos desta disciplina devem dominar os conteúdos abordados nas seguintes disciplinas:</p>

Figura 42 - Informação orgânica sobre uma disciplina

7.8.3 Informação de turmas

O objectivo deste componente é permitir aos utilizadores a consulta dos sumários e da lista de alunos das turmas a que está associado.

Este componente veio permitir que pela primeira vez na universidade, docentes e alunos tenham informação integrada sobre quem está inscrito numa turma (lista de alunos), com as respectivas fotografias e sumários.

Por exemplo, da disciplina de Análise Matemática I é possível consultar os sumários (Figura 43) já lançados via PACO (consultados por intermédio de web service), assim como consultar a lista de alunos de uma turma (Figura 44). Por questões de confidencialidade, a informação dos alunos está deliberadamente rasurada.

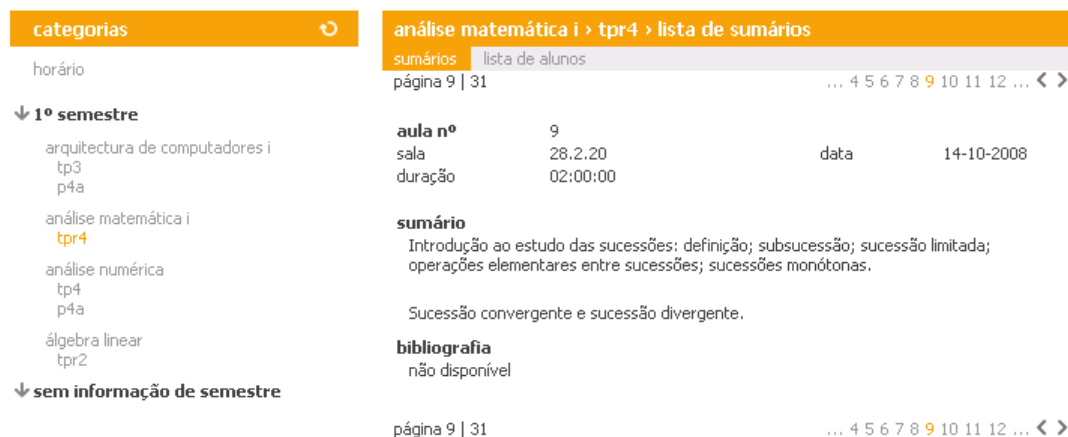


Figura 43 - Lista de sumários

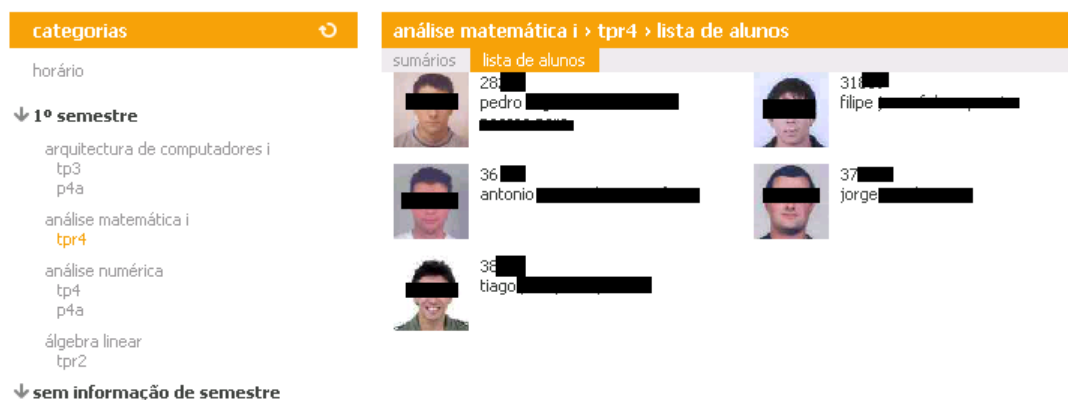


Figura 44 - Lista de alunos de uma turma

De referir que a nível da Universidade de Aveiro, este foi o primeiro sistema onde os utilizadores (discentes e docentes) puderam confirmar visualmente os alunos de uma turma.

7.9 Explorador de ficheiros da sua área pessoal

O objectivo deste componente é integrar o acesso ao ARCA na interface do my.ua.

Todos os utilizadores da Universidade têm uma área de trabalho pessoal no arquivo dos servidores centrais. Esta área (chamada de ARCA) é mapeada como um disco virtual sempre que um utilizador acede a um dos computadores existentes nos laboratórios da

Universidade com as suas credenciais. Com esta funcionalidade, o trabalho e progresso dos utilizadores nas disciplinas, não está restringido ao uso de um determinado computador. Poderá utilizar livremente os computadores que entenda, desde que tenha acesso aos mesmos e desde que os programas que necessita estejam instalados.

Um dos problemas desta área é que o seu acesso estava limitado a duas interfaces:

- local de trabalho nos computadores acedidos e
- interface web própria do sistema (<http://arca.ua.pt>).

A Figura 45 apresenta a interface de acesso à área pessoal de documentos. Nela é possível distinguir duas áreas distintas: uma secção de áreas autorizadas (à esquerda) e uma secção central.

Na secção de áreas autorizadas, dependendo das funções desempenhadas por um utilizador, este poderá ter acesso, além da sua área pessoal, a áreas partilhadas com outros utilizadores, mediante listas de autorização e controlo previamente definidas. Exemplo de tais situações é, por exemplo, uma área para documentos acessíveis por todos os docentes de um departamento. Note-se que não faz parte das capacidades deste componente a gestão das políticas de acesso e de permissões às diversas áreas; tal função é da responsabilidade do CICUA. O que este componente permite é, rapidamente, listar apenas e só as áreas em que o utilizador tem acesso (seja de leitura, seja de leitura e escrita).



Figura 45 - Interface acesso área pessoal de documentos

A secção central permite a consulta, download e upload de documentos, assim como a gestão de pastas e documentos (mover, copiar, renomear, apagar e criar).

7.10 Editor de páginas HTML da sua área web

O objectivo deste componente é potenciar a rápida edição de páginas HTML em linha.

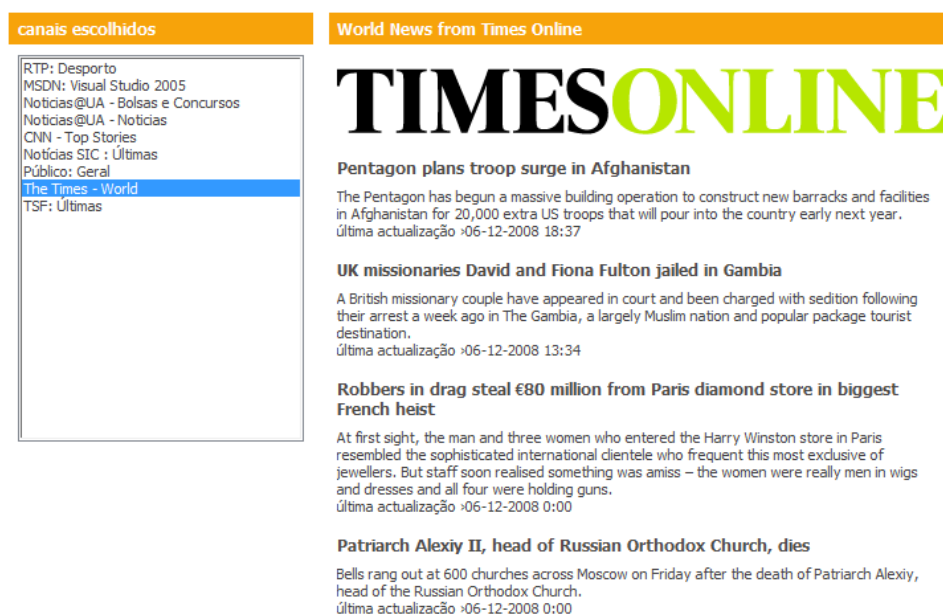
O componente de edição de páginas HTML está intrinsecamente ligado ao explorador da área pessoal, visto ser este que permite a navegação até aos conteúdos a editar. Uma das pastas especiais existentes na Arca é a que armazena a página pessoal (de responsabilidade pessoal) de cada utilizador. A possibilidade de se aceder através do my.ua a essa pasta, e ainda a possibilidade de edição da mesma, dão a este componente uma capacidade única e até agora não disponível. Actualmente, as alterações nas páginas pessoais são feitas recorrendo a editores externos com posterior actualização na pasta.

7.11 Leitor de conteúdos noticiosos preferidos

O objectivo deste componente é disponibilizar conteúdos noticiosos.

O componente de conteúdos noticiosos permite que os utilizadores seleccionem os canais RSS que pretendem ter disponíveis para consulta dentro do portal. Trata-se no fundo de um agregador de RSS integrado no portal, que permite a associação de canais de acordo com as preferências e gostos dos utilizadores.

A Figura 46 apresenta a interface do leitor. Na Figura 47 é apresentada a interface de gestão dos canais apresentados neste componente.



The screenshot shows a news reader interface. On the left, under the heading "canais escolhidos", there is a list of RSS channels: RTP: Desporto, MSDN: Visual Studio 2005, Noticias@UA - Bolsas e Concursos, Noticias@UA - Noticias, CNN - Top Stories, Noticias SIC : Últimas, Público: Geral, The Times - World (highlighted in blue), and TSF: Últimas. On the right, under the heading "World News from Times Online", the "TIMESONLINE" logo is displayed. Below the logo, three news articles are listed:

- Pentagon plans troop surge in Afghanistan**
The Pentagon has begun a massive building operation to construct new barracks and facilities in Afghanistan for 20,000 extra US troops that will pour into the country early next year.
última actualização :06-12-2008 18:37
- UK missionaries David and Fiona Fulton jailed in Gambia**
A British missionary couple have appeared in court and been charged with sedition following their arrest a week ago in The Gambia, a largely Muslim nation and popular package tourist destination.
última actualização :06-12-2008 13:34
- Robbers in drag steal €80 million from Paris diamond store in biggest French heist**
At first sight, the man and three women who entered the Harry Winston store in Paris resembled the sophisticated international clientele who frequent this most exclusive of jewellers. But staff soon realised something was amiss – the women were really men in wigs and dresses and all four were holding guns.
última actualização :06-12-2008 0:00
- Patriarch Alexiy II, head of Russian Orthodox Church, dies**
Bells rang out at 600 churches across Moscow on Friday after the death of Patriarch Alexiy, head of the Russian Orthodox Church.
última actualização :06-12-2008 0:00

Figura 46 - Lista de conteúdos noticiosos



Figura 47 - Gestão de canais noticiosos

7.12 Actualização de dados pessoais no serviço central

O objectivo deste componente é permitir a actualização de dados pessoais de forma célere.

A partir do momento em que os dados dos utilizadores estão armazenados de forma central (apesar de haver cópias locais em alguns sistemas), tornou-se importante a criação de um método que permitisse a qualquer utilizador alterar parte dos seus dados pessoais.

A Figura 48 mostra a interface de actualização de dados pessoais.



Figura 48 – Interface de actualização de dados

O my.ua é utilizado como “o primeiro serviço” a que um utilizador da UA tem acesso imediatamente após a sua matrícula. Através do portal e nomeadamente através deste componente pode alterar a sua morada móvel, o seu número de telefone, etc.

7.13 Alteração de palavra passe

O objectivo deste componente é permitir a alteração da palavra passe de um utilizador.

Dado o ambiente integrado de operação do portal com o restante sistema de informação, a alteração de palavra passe afecta todos os sistemas.

A alteração de palavra passe é central, pelo que todos os acessos a contas de correio electrónico, acesso a computadores, VPN, etc., serão afectados pela mudança.

7.14 Alteração da configuração das diversas páginas

O objectivo deste componente é permitir a personalização do arranjo gráfico das diversas operações nas páginas.

É da responsabilidade deste componente, garantir que algumas configurações impostas pela Universidade, não sejam alteradas pelo utilizador. Contudo, nem todos os componentes têm uma disposição obrigatória, pelo que nesses casos, o utilizador é livre de os movimentar e/ou retirar.

O componente apresentado na Figura 49 interpreta as restrições e possibilidades dadas no perfil do utilizador, em termos de áreas receptivas (na imagem, o Painel Esquerdo e Painel Direito) da página em questão (*default*), e componentes móveis (na imagem, todos à excepção do *RSS news reader* que ocupa o Painel Central).



Figura 49 - Configuração da disposição dos diversos componentes

De acordo com o definido no perfil do utilizador, os componentes poderão ainda ser ocultados ou tornados visíveis.

A Figura 50 mostra, parcialmente, a página de entrada após um rearranjo de alguns dos componentes móveis. Verifica-se pela figura que os dados das contas de correio electrónico foram trocados com a identificação do utilizador e que a agenda, que por omissão está na coluna da direita, foi colocada no fundo da coluna da esquerda.

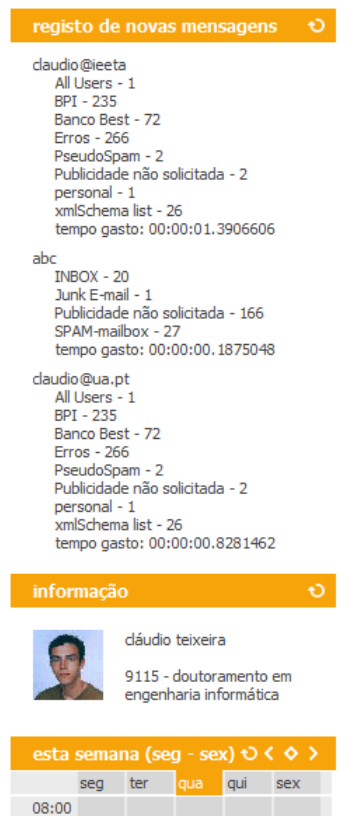


Figura 50 - Rearranjo da página de entrada

7.15 Sobre os componentes

A Tabela 6 apresenta um resumo das fontes de informação consultadas por cada um dos componentes desenvolvidos. Cada fonte de informação disponibilizou um conjunto de métodos que permitem a consulta da informação pretendida. A definição, o desenvolvimento e a disponibilização dos métodos é da responsabilidade do próprio serviço. Em alguns casos, a comunicação é feita por intermédio de web services, ao passo que noutros é feita recorrendo a documentos em formato RSS.

De acordo com o dinamismo, abstracção e segurança pretendidos para todo o sistema, todos os componentes aqui referidos só estão disponíveis a um determinado utilizador no caso do perfil deste o contemplar.

Estes componentes foram desenvolvidos tendo em conta a arquitectura da aplicação. No entanto, provam o conceito, o que permitirá que novos componentes venham a ser desenvolvidos no futuro.

Quadro resumo de componentes e fontes de informação	Serviços académicos (Web-service)									
	Serviços académicos (Web-service)	Arca (Sistema de ficheiros)	uaOnline (RSS)	Servidores RSS (RSS)	Servidor de agenda (iCal)	Servidor de correio electrónico (IMAP4)	Gestão documental (Web-Service)	Portal Acesso (Web-Service)	RCU (Web-Service)	Gestão de perfis (Web-Service)
Leitor de notícias			X							
Informação pessoal dos serviços académicos	X									
Aulas de hoje	X									
Cliente de correio electrónico e agenda					X	X				
Mensagens gestão documental							X			
Sumário de mensagens de correio electrónico						X				
Agenda pessoal			X		X					
Horário lectivo	X									
Informação disciplinas	X							X		
Informação de turmas	X									
Explorador de ficheiros da sua área pessoal		X								
Editor de páginas HTML da sua área web		X								
Leitor de conteúdos noticiosos preferidos				X						
Actualização de dados pessoais no serviço central									X	
Alteração de palavra passe									X	
Alteração da configuração de diversas páginas										X

Tabela 6 - Quadro resumo de componentes e fontes de informação

A criação de novos componentes deverá ser assegurada pelos fornecedores de serviço. Para tal, foi criado um componente base que contém toda a lógica de integração com a plataforma do portal. Caberá ao fornecedor de serviço criar a lógica de serviço necessária sobre este componente.

8 Resultados

Um dos resultados para que o my.ua contribuiu significativamente foi na alteração da filosofia de lidar com a informação na Universidade. O sistema de informação da universidade sofreu alterações, como é perceptível quando se compara o esquema da Figura 19 (sistema antigo) com o esquema da Figura 51 (sistema actual).

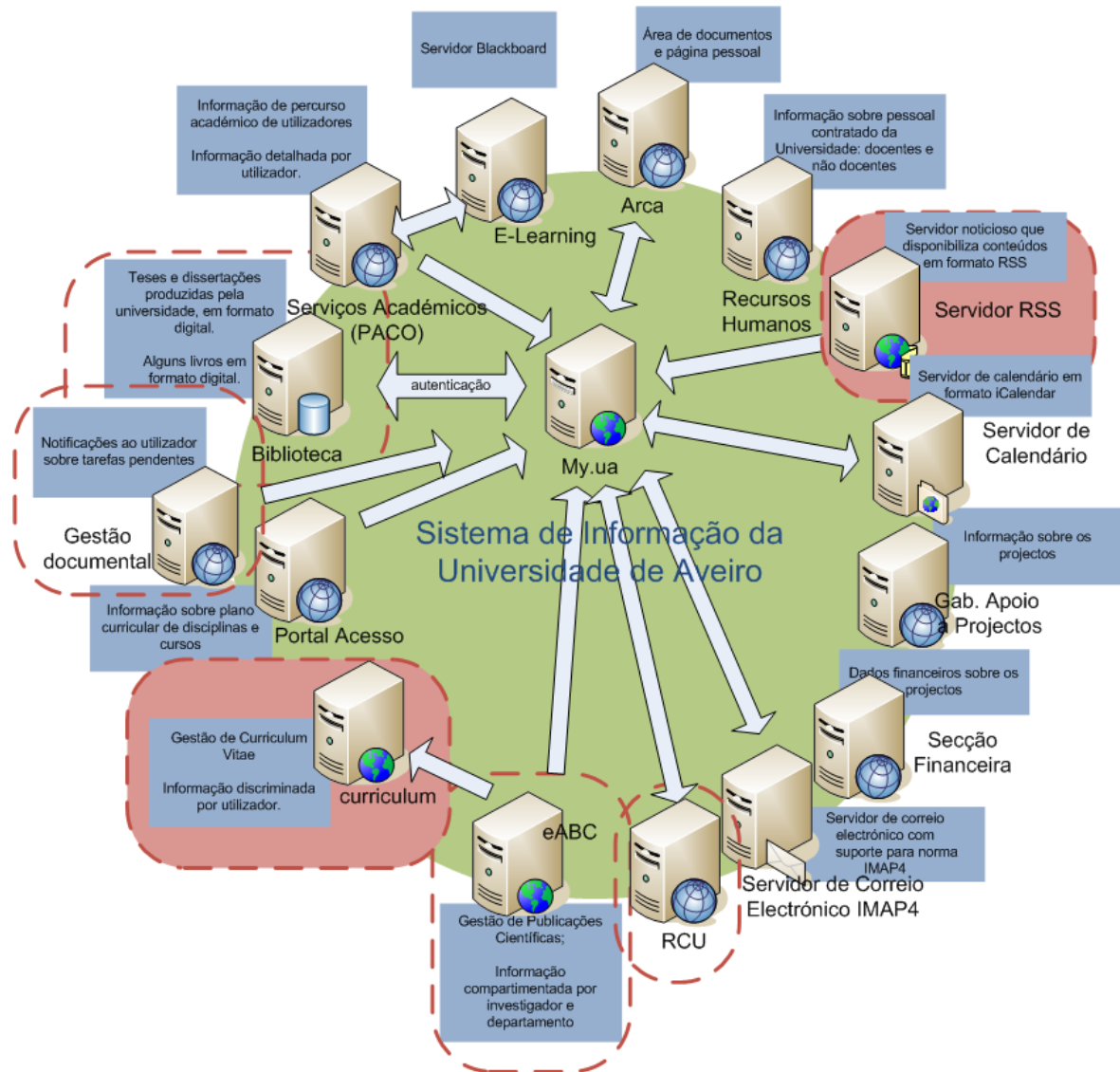


Figura 51- Sistema de informação pós my.ua

Verifica-se na Figura 51 que há interações entre sistemas apenas num sentido (leitura ou apresentação de informação) e há interações em ambos os sentidos (leitura e alteração de informação). Além disso, o PACO interligou-se com o e-Learning, permitindo uma navegação contextualizada. No caso do RCU, por ser um sistema usado por praticamente

todos os restantes, optou-se por simplificar a sua interacção, mantendo apenas as trocas de informação com o my.ua.

Estando o portal em exploração há mais de dois anos, é agora possível analisar estatisticamente alguns elementos, como:

- Receptividade e uso por parte dos utilizadores
- Usabilidade associada a toda a lógica do sistema
- Performance da gestão de informação em XML

8.1 Receptividade e uso – dados estatísticos

Como forma de monitorar a evolução do sistema recorreu-se ao serviço de estatísticas do Google Analytics (Google, 2008). O sistema foi colocado oficialmente em produção em 16 de Dezembro de 2006. Assim, serão analisados os dados referentes aos anos de 2007 e 2008.

Dos vários elementos estatísticos recolhidos, serão objecto de análise:

1. o número de visitas, ou sessões únicas no sistema,
2. o número de páginas vistas por visita e
3. o tempo médio de visita.

Um dos parâmetros que permite avaliar o interesse num site por parte dos utilizadores é o número médio de páginas vistas em cada sessão de utilização ou visita.

Por visita entende-se o período que medeia desde que o utilizador acede ao site até à última página pedida ao servidor.

O número de visitas reflecte de certa forma a popularidade e/ou interesse dos utilizadores no sistema. Um sistema pode ter muitos visitantes pelo facto de haver curiosidade nos serviços por este disponibilizados, mas se não houver algo que os fidelize, estes, muito certamente, não regressarão ao sistema.

Se o número médio de páginas vistas for elevado, significa que houve lugar a diversas interacções com o utilizador, o que poderá significar um maior consumo de informação orientada às particularidades e preferências do utilizador em questão.

Contudo, se o número médio de páginas por sessão for elevado e o número de visitas for diminuindo, provavelmente estamos perante um de dois cenários:

1. Desagrado por parte dos utilizadores, visto que apesar de terem tentado encontrar a informação que pretendiam, esta não estava acessível de uma forma clara (daí o número de páginas vistas ser elevado). Se os utilizadores não conseguem encontrar a informação pretendida, o mais provável será abandonarem o sistema em questão em detrimento de outros com interfaces mais acessíveis.
2. O conteúdo do site é muito específico e interessante mas a sua variedade é pouca, o que implica que depois de visto uma vez, não há novidade na próxima visita.

Se, pelo contrário, o número de visitas for aumentando então significa que muito provavelmente a taxa de fidelização dos utilizadores está a aumentar.

8.1.1 Visitas

A Figura 52 mostra um gráfico com a evolução estatística do sistema, em termos do número de acessos ao longo dos dois primeiros anos de existência do my.ua.

O aspecto que mais sobressai do gráfico é a tendência ascendente do número de visitas. Se durante os meses iniciais do my.ua, o sistema passou algo despercebido, não ultrapassando as 10.000 visitas num mês, a partir de Setembro do ano seguinte os valores dispararam, voltando a disparar, novamente, em Setembro de 2008.

Os factores que poderão justificar este comportamento são, em primeiro lugar, a publicidade feita ao sistema, em segundo, as datas chave de início de ano lectivo e finalmente a fidelização dos utilizadores. Note-se que no arranque do ano lectivo, estes foram os primeiros alunos a terem o my.ua como “sistema de entrada” na universidade.

Como explicado anteriormente, a visão do my.ua ser apenas de leitura de todo o sistema, foi evoluindo para um cenário mais interactivo, com os responsáveis dos sistemas envolvidos a passarem mais responsabilidades para o campo do my.ua. Uma das responsabilidades entregues ao my.ua, foi o registo e personalização do Utilizador Universal – a identificação única para todos os sistemas da universidade. Foi, de certa forma, com alguma naturalidade que esta tarefa foi enquadrada no my.ua, visto ser este o sistema encarregue de disponibilizar interfaces personalizadas aos utilizadores. Com esta tarefa, o my.ua passou a ser um dos primeiros sistemas informáticos apresentados a novos alunos e elementos da universidade. Ainda que numa base de imposição, todos sem

excepção, teriam que passar pela escolha e activação do utilizador universal por intermédio do my.ua.

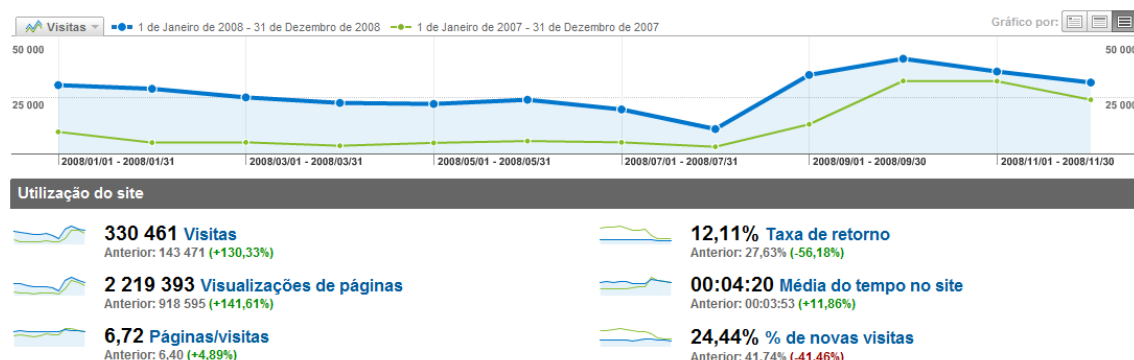


Figura 52 - Gráfico comparativo do número de visitas nos anos de 2007 e 2008

Apesar de o sistema estar anunciado de forma oficial em todos os sites da universidade, a sua presença foi pouco notada durante os primeiros meses de actividade. Além disso, o número de acessos foi diminuindo, baixando cerca de 50% se comparado com o pico de utilização até então. Apesar de, nos primeiros meses após o lançamento do my.ua o número de acessos ter baixado, notam-se duas tendências no comportamento dos utilizadores: a curiosidade e o uso. No primeiro ano de produção, o número de visitas ao portal foi diminuindo desde Janeiro até Maio. Desde esse momento, a tendência de utilização foi crescendo. Este factor pode ser revelador de um maior interesse e necessidade de usar o portal. Note-se ainda que Maio é o fim do ano lectivo, pelo que se poderá conjecturar uma maior utilização por parte dos docentes, devido às funcionalidades de visualização de fotos dos alunos na altura da submissão de notas da avaliação contínua. Tal hipótese é aparentemente suportada pelas visualizações de páginas: durante o mês de Maio, a parte das aulas é responsável por 8,41% de páginas vistas, o que contrasta com os 4,44% de média do primeiro ano. Outro aspecto a tomar em consideração é o aumento consecutivo após o mês de Setembro (data de inscrições na Universidade): este aumento mostra algum grau de fidelização no sistema – os utilizadores recém-chegados estão a adoptar o my.ua como ferramenta do seu dia-a-dia.

8.1.2 Páginas vistas

A Figura 53 mostra uma análise comparativa do número de páginas vistas nos anos de 2007 e 2008. Neste gráfico, nota-se um aumento substancial do número de páginas vistas

durante os meses de Setembro e Outubro, resultado directo das inscrições de novos alunos na Universidade.

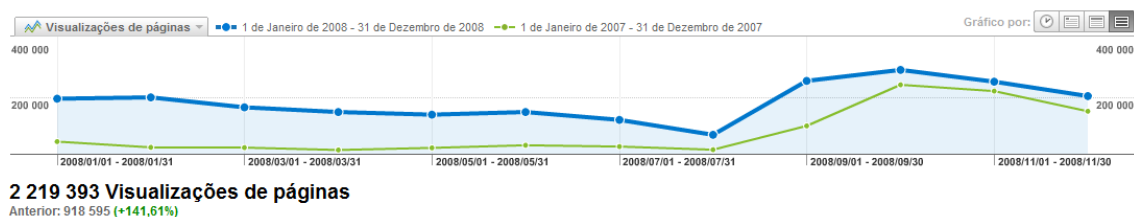


Figura 53 - Gráfico comparativo do número de páginas vistas nos anos de 2007 e 2008

De realçar também o aumento de páginas vistas em 2008. Os mais de dois milhões de páginas vistas representam um aumento de mais de 140% do número de páginas vistas no ano anterior (918 595).

8.1.3 Páginas por visita

A Figura 54 mostra uma análise comparativa do número de páginas por visita entre o ano de 2007 e 2008. Como é visível na figura, nos primeiros meses de produção, os utilizadores que acederam ao sistema navegaram em média por cerca de 4 páginas em cada visita.

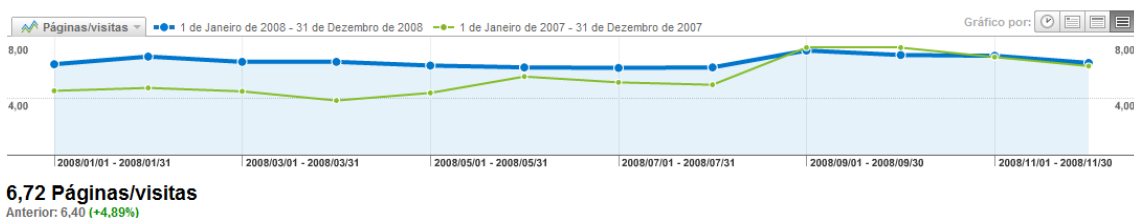


Figura 54 - Gráfico comparativo de páginas por visita nos anos de 2007 e 2008

A partir do mês de Maio do primeiro ano, o número médio de páginas por visita foi aumentando, o que se traduz numa maior utilização do sistema por parte dos utilizadores.

Importa acrescentar que a partir de Setembro de 2007, o número de páginas por visita manteve-se praticamente constante nas cerca de 7 páginas por visita, com ligeiras flutuações (entre 6,1 e 7,4). Este cenário revela que o sistema estabilizou em termos de utilização. Como não houve adição de novas funcionalidades ao longo deste período, os utilizadores não sentem necessidade de um maior número de interações com o sistema. Este valor só poderá crescer através da adição de novos serviços com real valor para os utilizadores.

8.1.4 Tempo médio por visita

O tempo que um utilizador despende numa visita a um sistema é também ele relevante da utilidade do mesmo para as tarefas que pretende resolver, principalmente se o número de acessos ao sistema não for diminuindo.

O gráfico na Figura 55 revela um aumento do tempo médio de acesso ao sistema durante o primeiro ano de exploração do portal. No ano seguinte, o tempo médio oscila entre os 3m41s e os 5m07s. Novamente, estes valores indicam uma utilização mais presente, com maior disponibilidade temporal para ver e/ou interagir com os conteúdos disponibilizados pelo portal.

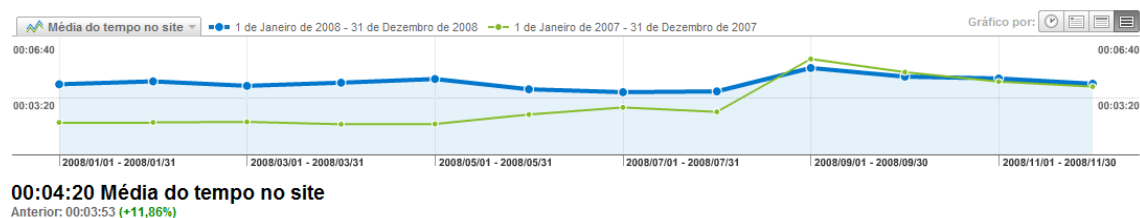


Figura 55 - Gráfico comparativo do tempo médio por visita nos anos de 2007 e 2008

Tal como no caso das páginas por visita, este tempo apenas tenderá a aumentar através da adição de novos serviços.

8.1.5 Páginas mais visitadas

Atendendo a que se pretende que o my.ua seja um portal personalizável pelos utilizadores, importa mencionar quais as páginas mais visitadas, assim como o grau de interacção que os utilizadores têm com elas.

A Figura 56 apresenta os serviços mais visitados pelos utilizadores, assim como a quantidade média de interacções com o serviço ou página em questão.

No seu conjunto, estes acessos representam mais de 97% dos acessos ao sistema.

As páginas mais visitadas são a página de correio electrónico, a página de login e a página inicial. De acordo com o perfil de cada utilizador, a informação nas páginas poderá variar, assim como a necessidade/vontade de interagir com esta.

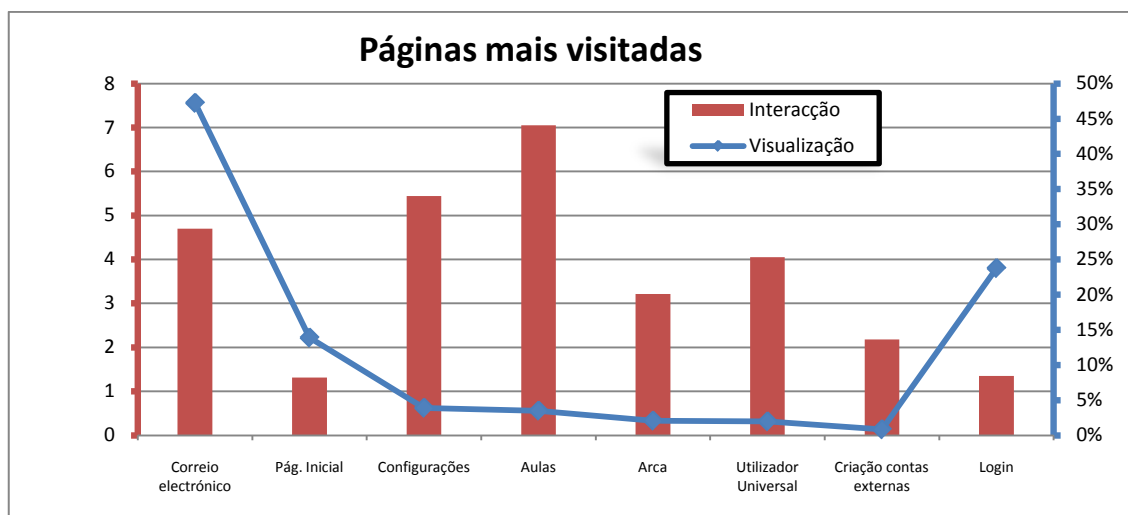


Figura 56 - Páginas mais visitadas e interações

No entanto, do gráfico é possível verificar que apesar da área de aulas ser das menos visitadas, quem a visita navega na informação que esta contém: turmas, sumários e informação sobre uma disciplina. Na vertente oposta, a página inicial, por tipicamente ter diversas fontes de informação, mas todas apenas de visualização, tem em média menos de duas interações. No caso do correio electrónico, a média de cinco interações permite ao utilizador consultar e enviar cerca de duas mensagens. Um dado também interessante é o número de interações presente na secção de configurações, o que poderá sugerir alterações explícitas de perfis de utilizador.

A “Criação de contas externas” presente no gráfico refere a área para criação de contas de utilizadores não associados à Universidade, mas que podem ter interesse no acesso personalizado aos serviços públicos (alunos prospectivos, jornalistas, etc.). A secção “Utilizador Universal” refere a área para activações de contas de utilizadores da Universidade.

8.1.6 Dispersão geográfica

Como seria de esperar, a maior parte dos acessos é de âmbito nacional (97%), contudo, os acessos a partir de Aveiro (incluindo acessos na Universidade) contabilizam apenas 41,3% dos acessos, o que significa que, para além de ser uma ferramenta para uso durante os períodos de aulas e trabalho no campus, os utilizadores acedem também ao sistema fora deste meio.



Figura 57 - Dispersão geográfica de acessos ao my.ua

A Figura 57 identifica as zonas de Portugal Continental de onde foram efectuados os acessos. De notar que qualquer acesso a sistemas online irá produzir como localização geográfica uma versão aproximada da localização real, baseada nos pontos de acesso à rede registados pelo operador. Por este motivo, o mapa apresentado deverá ser visto como um auxílio para se perceber quais as zonas responsáveis (sem elevada especificação de zona) pelo maior número de acessos.

Assim, pode-se concluir que Aveiro, Lisboa e Porto (por esta ordem) são as zonas de onde é oriunda a maior parte dos acessos ao portal.

8.2 Testes de usabilidade

Como referido, o portal foi oficialmente lançado em 16 de Dezembro de 2006, por ocasião do 33º aniversário da Universidade de Aveiro. Nessa data, foram também apresentados outros sistemas da universidade, entre eles, a nova plataforma para gestão de sites departamentais e da universidade (Casanova, Pinto, Teixeira, & Monteiro, 2007).

Aproveitando o lançamento de todos estes novos sistemas, foi proposto aos alunos da disciplina de Interfaces Humano-Computador (leccionada na altura ao 3º ano da Licenciatura de Eng.^a de Computadores e Telemática da Universidade de Aveiro) a análise crítica das questões de usabilidade existentes em alguns dos sistemas apresentados. Desse trabalho resultaram duas publicações em que são referidos os métodos de avaliação e

análise de alguns dos sistemas e os resultados alcançados (Santos et al., 2007; Santos et al., 2008).

Um dos sistemas avaliados no estudo foi o portal my.ua, em duas componentes: o portal na sua generalidade e o componente de correio electrónico. Esta divisão ficou a dever-se à complexidade das tarefas em questão e da dificuldade na sua avaliação simultânea. O componente de correio electrónico, pela quantidade de funcionalidades que possui, foi naturalmente escolhido como um subsistema que deveria ser avaliado de forma independente.

Em (Santos et al., 2008) os autores referem quais os erros mais graves encontrados na avaliação do portal e do componente de correio electrónico. Sugerem também formas de corrigir as falhas apontadas. De referir que a maioria dos problemas apontados são de reduzida gravidade (do ponto de vista de funcionalidades) e relacionados com questões puramente de interface gráfica e usabilidade.

De entre os erros apontados, destacam-se:

- A opção na agenda pessoal pelo arranjo gráfico horizontal dos eventos com duração igual ou superior a um dia
- Inexistência de legenda das cores utilizadas para diferenciar os diversos eventos na agenda;
- Existência de campos não assinalados como obrigatórios;
- Impossibilidade de recuperação de palavra passe;
- Iconografia diferente da habitualmente usada em sistemas análogos;
- Configuração da conta de correio electrónico.

Os problemas não relacionados com a componente gráfica (como a configuração da conta de correio electrónico) foram resolvidos de imediato, após o conhecimento dos resultados desta avaliação.

As questões gráficas não foram resolvidas, visto que o aspecto gráfico não é parte integrante das responsabilidades atribuídas à equipa de desenvolvimento do sistema, sendo esta da inteira responsabilidade da equipa de design contratada pela Universidade para o efeito.

8.3 Gestão de informação armazenada em XML

Em (C. Teixeira, Pinto, & Martins, 2008b) é apresentado um estudo de laboratório sobre a performance associada à pesquisa de informação que é armazenada em XML. Trata-se de um estudo cujo objectivo foi determinar qual das tecnologias disponíveis na altura daria a melhor resposta para um problema por vezes difícil de solucionar: procurar informação armazenada em XML.

Tal como referido anteriormente, existem várias opções para manusear informação estruturada em XML. O problema é saber qual das opções é a mais válida para um cenário em particular. Como no sistema de gestão de perfis do my.ua, a taxa de pesquisa de informação sobre um determinado perfil é bastante superior à taxa de actualização do perfil (um perfil é alterado esporadicamente), foi criado um cenário que permitisse efectuar experiências controladas sobre a pesquisa de informação previamente armazenada. Isto significa que, o estudo realizado, não focou questões relacionadas com a performance do armazenamento e indexação de informação, preocupando-se apenas com a performance associada à pesquisa.

Este cenário foi baseado na Wisconsin Benchmark (DeWitt, 1993), tendo utilizado como documentos de teste 40.000 documentos com informação UNIMARC (IFLA, 2000), armazenada segundo a norma UNIMARC-XML (Carvalho, 2005). Os documentos em questão foram também gerados de acordo com as recomendações da Wisconsin Benchmark.

Como opções ao armazenamento foram utilizadas aplicações, quer de código-fonte aberto, quer aplicações comerciais. As aplicações testadas foram Sedna, Lucene, Index Server e Sql Server 2005.

A Sedna (MODIS: ISPRAS, 2008) é uma NXD, que suporta o processamento de XML usando o standard XQuery. A versão usada é a 2.2.154.

A Lucene (Apache Software Foundation, 2007) é uma aplicação de indexação de conteúdos, originalmente escrita em Java. Possui também uma versão .NET que resultou de uma migração classe a classe e algorítmica da versão Java. A versão utilizada é a Lucene.Net 2.0 “final” compilação 004.

O Index Server é um utilitário disponibilizado nas versões do Windows (XP e 2003) para indexação de conteúdos. Será usado como motor externo para indexação e pesquisa de ficheiros XML directamente no sistema de ficheiros.

O SQL Server 2005 é uma XEnDB, suportando o tipo XML em colunas de tabelas relacionais e permitindo operações XQuery e XUpdate sobre as mesmas. Suporta também indexação de expressões XPath, seja por valor seja por propriedade. Este SGBD foi usado para testes em cenários bastante distintos: armazenamento directo de XML num registo e armazenamento relacional (interpretando e decompondo o documento XML em tabelas).

Foram realizadas pesquisas sobre os documentos armazenados, que contemplavam desde a simples pesquisa por nome de autor, até pesquisas que teriam como resultado uma determinada percentagem de registos (pesquisas de base percentual – PB Query), estando os resultados apresentados na Tabela 7.

Foram realizados as seguintes pesquisas:

- QR1 – Pesquisa por nome de autor. Esta pesquisa serviu acima de tudo para garantir que as pesquisas seguintes teriam já todo o sistema preparado para responder aos pedidos. Normalmente, existe um overhead associado à abertura da ligação com o motor de base de dados. Para que essa despesa extra não influenciasse os resultados pretendidos, foi feita esta pesquisa “pré-teste”.
- QR2 – Pesquisa por nome de autor (com nome distinto de QR1).
- PB x% – Pesquisa que devolve uma determinada percentagem (x%) de registos.
- QR 1 Elem – Pesquisa que devolve sempre apenas um registo, único na base de dados.

A Tabela 7 mostra os tempos (em segundos) que as diferentes pesquisas demoraram a devolver a informação em causa, assinalando para cada caso o(s) sistema(s) com tempos mais baixos.

Uma rápida análise à tabela revela que a NXD presente nunca tem o melhor tempo, possuindo ainda quatro dos piores tempos. Por outro lado, a alternativa Lucene.Net apresenta três melhores tempos.

Todas as pesquisas com um factor de selectividade superior a 2%, foram mais rápidas recorrendo a SQL relacional, enquanto para selectividades inferiores a 2%, estas foram

mais rápidas usando Lucene.Net. A exceção a esta regra são as pesquisas QR1 e QR2 (devolvem 10 autores cada, o que está dentro do 2% do Lucene.Net) que, pela sua complexidade (pesquisa de nome em dois sub-elementos por cada autor) foram mais rápidos usando as potencialidades XEnDB do SQL Server.

	QR1	QR2	PB 50%	PB 2%	PB 1%	PB 0.001%	QR 1 Elem
Sedna	4,65	2,85	73,27	46,51	38,97	1,33	1,26
Index Server	0,68	0,46	70,04	2,47	1,33	0,09	0,09
SQL Relacional	2,08	0,63	2,78	0,33	0,33	0,27	0,28
SQL XML	0,34	0,25	76,58	6,07	4,57	1,58	1,81
Lucene.Net	0,62	0,25	9,86	0,46	0,21	0,06	0,08

Tabela 7 - Tempos de pesquisa (em segundos)

A solução de sistema de ficheiros, em conjunto com o Index Server, apresenta-se como uma solução competitiva para pesquisas que devolvam poucos resultados.

À exceção das pesquisas QR1 e QR2, a pesquisa de informação revelou-se mais rápida usando repositórios que não trabalham directamente com XML: o SQL relacional decompôs a informação para um modelo relacional e o repositório Lucene.Net indexa informação em pares chave/valor, independentemente da sua estrutura original.

Este estudo serviu acima de tudo para demonstrar que não existe uma solução global/única capaz de responder a esta questão. Se, em regra, as pesquisas devolverem conjuntos de resultados relativamente grandes (em função do universo de valores), então uma base de dados relacional será provavelmente a melhor candidata; caso o conjunto de resultados seja relativamente pequeno, então a indexação será provavelmente a melhor abordagem.

Uma das características do Lucene, é a sua capacidade de manter indexados documentos já eliminados. Este sistema não tem uma gestão própria da latência da informação. Ou seja, não há a garantia de correspondência documento/indexação, se não se tiver em consideração mecanismos de notificação de alteração de ficheiros no sistema operativo. A indexação da informação via Lucene é completamente controlada por um programa feito à medida que retira do documento XML a informação a indexar e a envia para o motor de indexação. Caso a alteração de documentos passe ao lado deste programa, então todo o sistema fica em risco.

No caso particular do sistema de gestão de perfis, a operação mais recorrente será a pesquisa e recuperação do perfil de um utilizador em particular. Neste caso o cenário é em tudo semelhante ao “QR 1 Elem”, apresentado na Tabela 7. Neste cenário, além do Lucene, o Index Server tem um comportamento bastante aceitável, sendo o 2º melhor tempo. No caso do Index Server, a baixa latência da informação constante no índice, é assegurada de forma interna. A monitorização de pastas com documentos é automática, pelo que qualquer alteração aos documentos será reflectida nos termos indexados. Ao contrário do Lucene, no caso de se apagar um documento, a informação indexada sobre este é também eliminada. Se, no caso do Lucene, é necessário um programa feito à medida para alimentar o índice, no caso do Index Server, e por este não suportar de forma nativa a indexação de documentos XML, é necessário uma extensão externa que lhe permita interpretar o seu conteúdo e definir os elementos a indexar.

Como referido anteriormente, não existe nenhuma solução completa e, no caso do sistema de gestão de perfis, pode-se até falar de duas soluções bastante equiparadas. Optou-se pela solução Index Server, por garantir de forma autónoma toda a monitorização necessária. A relativamente baixa performance em pesquisas que devolvam mais que um resultado será apenas sentida em cenários de administração da plataforma, pelo que serão de impacto reduzido.

9 Conclusões

Quando se implementa um novo serviço, é necessário ponderar qual o impacto do aumento do números de utilizadores e qual o impacto do novo serviço nos serviços já existentes. Trata-se portanto, de verificar a escalabilidade do novo sistema e a manutenção da performance e disponibilidade dos sistemas por este solicitados.

O sistema my.ua não tem um impacto significativo na performance do servidor em que está alojado. Trata-se de um sistema que tem, neste momento, cerca de 1000 utilizadores por dia, o que apesar de ser um valor encorajador para um sistema cujo propósito é servir os utilizadores associados à Universidade de Aveiro, não é um valor elevado em termos de sistemas de informação. De entre os diversos factores que contribuem para o baixo impacto do sistema no servidor, pode-se realçar o facto de o portal servir apenas de agregador de informação, consultando em tempo real os diferentes sistemas fornecedores de informação. O processamento de dados, em si, não é significativo.

Tendo em consideração que o my.ua consulta informação existente em diversos serviços, é de esperar um acréscimo de tráfego para estes. Contudo, atendendo a que a informação é pesquisada apenas nos sistemas a que o utilizador já tem acesso, o acréscimo de tráfego será de certo modo proporcional à visibilidade do serviço perante o utilizador, ou seja, quanto maior o número de utilizadores associados a um serviço da universidade, maior será o tráfego direccionado para o mesmo. A título de exemplo, as notícias da página inicial implicam por si só um aumento de tráfego do uaonline inferior a 10% (em relação ao tráfego gerado pelo próprio site uaonline), neste que é um dos sistemas mais usados na universidade.

9.1 *Sobre os objectivos do sistema*

O objectivo central desta tese é demonstrar a viabilidade de uma arquitectura de portal Web, com integração de múltiplos serviços, numa única interface centrada no utilizador.

A Universidade de Aveiro foi usada como caso de estudo, tendo sido demonstrado que é possível integrar este sistema noutros cenários.

Foi criada e apresentada toda a lógica inerente à gestão de perfis e utilizadores, assim como os diferentes componentes desenvolvidos, numa perspectiva funcional e de viabilização do projecto em si.

Com a entrada em funcionamento do my.ua, não foi substituído nenhum serviço ou sistema de informação, tendo sido criados novos canais de comunicação de informação com estes, numa perspectiva de maximizar a visibilidade da informação existente e, consequentemente, dos serviços a ela associados.

Foi apresentada uma proposta inovadora para um sistema de gestão de perfis de utilizador, tendo sido comparada com os modelos apresentados no estado de arte.

Foi também apresentada a estrutura e organização dos perfis de utilizador e das definições de perfis base.

Foram apresentados os diversos paradigmas associados a autenticação e SSO.

Foram discutidos e propostos meios de armazenar e gerir a informação produzida pelo sistema. O armazenamento de informação baseado em ficheiros XML guardados directamente no sistema de ficheiros, é um processo simples, mas o sistema de ficheiros permite uma indexação eficaz e pesquisas consideravelmente rápidas. Além disso, é um modelo facilmente escalável e redundante, mediante soluções para armazenamento de informação eficazes e transparentes, do ponto de vista lógico, para a aplicação: desde RAID até sistemas de armazenamento em rede (SAN – *Storage Area Network*).

Foi usado um servidor de modelos de utilizador, não para construir interfaces adaptáveis, mas para personalizar o acesso a serviços dentro de um ambiente corporativo, baseado em cargos e funções dinâmicas. Para tal, foi proposto o uso de perfis de utilizador que herda as suas características iniciais de perfis base, compostos por conjuntos transaccionais, regras de segurança e propostas de interface, onde a reavaliação do perfil é feita usando regras e processos formais definidos, sendo possível a actualização directa ou indirecta de perfis.

O sistema de gestão de perfis apresentado difere dos sistemas referidos no estado da arte. Trata-se de um sistema independente das aplicações, que pode extrair, inserir ou actualizar características dos modelos, seja num modelo em particular ou num conjunto de modelos. Sendo um servidor de modelos independente do sistema de informação, e tendo uma única interface entre o servidor e as aplicações, operações de manutenção e actualização de

servidor e de aplicações não comprometem a globalidade do sistema. Este sistema não pode viver sozinho, visto que necessita de um sistema de informação para gerir as regras de criação e atribuição dos perfis de utilizador e perfis base. Uma vez que este sistema foi criado para uso em organizações, não se considera que tal seja um factor de preocupação.

No que concerne à troca de informação entre sistemas, no acesso a informação pessoal/confidencial e na alteração de informação, é possível que cada componente defina quais as políticas de segurança exigidas para permitir a operação pretendida.

Apesar do sistema de gestão dos perfis armazenar a informação em ficheiros XML não cifrados, tal não é relevante do ponto de vista de segurança, visto que não são armazenados dados confidenciais do utilizador no seu perfil. Além disso, mesmo que lhe sejam adicionadas de forma manual permissões que este não possua, essas permissões serão revogadas no momento em que este tente o acesso ao serviço não autorizado. Do ponto de vista do portal, apenas é garantida a autenticação e validação do utilizador, sendo da responsabilidade de cada operação, a autorização de acesso às suas funcionalidades.

O sistema apresentado responde às questões iniciais e cumpre as funções para que foi proposto: do ponto de vista do utilizador, é uma montra de serviços a que este tem acesso, enquanto do ponto de vista da universidade é mais um meio de comunicação com os seus utilizadores. Do ponto de vista de arquitectura, todo o modelo proposto é não intrusivo, completamente independente do restante sistema de informação, estando adaptado para diferentes cenários.

9.2 *Perspectivas de evolução*

Tratando-se de um sistema em produção, a sua evolução é algo natural, que flui de acordo com as reais necessidades dos utilizadores e da instituição.

De acordo com a avaliação de usabilidade, os principais problemas encontrados pelos utilizadores dizem respeito a pormenores de interface, nomeadamente iconografia e realce de informação.

Tendo em consideração a adesão substancial por parte dos utilizadores, deverá ser ponderada a criação de mais componentes específicos para determinados públicos alvo, para assim aumentar a capacidade do sistema em produzir *apenas* a informação certa para a audiência certa. Além disso, os componentes existentes poderão ser alvo de refinamentos

e melhorias. A título de exemplo, uma das possibilidades de melhoria do componente de horário lectivo, passa pela transformação do número do edifício, andar e sala (ex. 11.1.27) numa representação georreferenciada do departamento, em conjunto com informação sinalética de como chegar à sala em questão. Essa informação já existe num sistema da Universidade, sendo portanto necessário criar um novo canal de comunicação.

O componente de correio electrónico, pela sua elevada popularidade, poderá também ser melhorado do ponto de vista de interface.

Do ponto de vista de desempenho, uma vez que os sistemas de gestão de bases de dados estão em evolução constante, é provável que testes futuros, de acordo com o modelo de avaliação de tempos de resposta apresentado, apresentem resultados distintos dos disponíveis actualmente. Poderá ser interessante nessa altura, alterar o modelo de gestão de informação em XML, colocando-a sob novo formato ou estrutura de dados.

Do ponto de vista de arquitectura, o sistema está também em aberto, pelo que poderá sofrer alguns ajustes mediante situações e necessidades não previstas até então. No sistema de gestão de perfis, um dos aspectos a melhorar será a sintaxe de suporte às regras dos processos formais. Este, deverá ser capaz de interagir sobre diferentes repositórios de forma transparente, com expressões complexas que interliguem diversos repositórios de dados.

9.3 Considerações finais

No decorrer dos trabalhos, e de modo a cumprir os objectivos traçados, houve alguns contratempos não antecipáveis, nomeadamente ao nível da própria arquitectura geral do sistema de informação da universidade e ao nível do relacionamento com os diferentes parceiros do projecto.

O sistema de informação da universidade antes deste projecto era, no fundo, um conjunto de ilhas de informação, muitas delas não comunicáveis. Este era, sem dúvida, um obstáculo para o my.ua, porventura o primeiro e que necessitava de uma resolução rápida, que não prejudicasse o desenvolvimento do portal. Foi na altura decidido superiormente que os sistemas da universidade deveriam partilhar informação entre eles, garantindo no mínimo o uso da identificação única que seria criada. Surgiram assim o Registo Central de Utilizadores (RCU) e o Utilizador Universal. Por ser um projecto de vital importância para

o sucesso do my.ua, o RCU foi acompanhado de muito perto, sendo a especificação inicial deste, fruto de estreita colaboração entre os responsáveis pelo my.ua e os responsáveis pelo Gabinete de Apoio à Gestão de Informação.

A confiança dos parceiros deste projecto (todos os sistemas da universidade) teve que ser conquistada de forma segura, provando em cada passo que a cooperação seria benéfica para todos os envolvidos, e que a confidencialidade, integridade e segurança dos dados próprios de cada sistema estava salvaguardada. Foi possível com a experiência, aprender como conseguir chegar a entendimentos e pontos de concordância, cedidos por ambas as partes. Todos os sistemas envolvidos são sistemas em produção, usados por mais de 13.000 utilizadores diariamente. Se os seus responsáveis fossem chamados para reunir com um recém-licenciado que lhes dissesse que pretendia construir um sistema para substituir todas as interfaces dos sistemas em produção, provavelmente o sistema estaria destinado ao fracasso antes mesmo de começar. Foi necessário dar um passo de cada vez; os responsáveis pelos serviços tiveram que ganhar confiança, tanto no sistema em desenvolvimento, como nos responsáveis por ele. No fundo, para a execução deste trabalho houve muita aprendizagem de relacionamento com equipas e serviços com objectivos distintos (e por vezes contrários), para além de todo o estudo e trabalho necessário para propor e demonstrar um sistema funcional.

Por ser necessário ganhar a confiança dos responsáveis, todos os componentes desenvolvidos na fase inicial eram apenas de leitura, coligindo a informação em tempo real das fontes. No entanto, após a entrada em produção do portal e explicação dos diferentes níveis de segurança previstos, houve um aumento da confiança no portal e na equipa responsável, pelo que foram criadas condições para o aparecimento de componentes que tenham de facto um impacto no sistema de informação. Componentes como inscrição em exames, publicação de notas, lançamento de sumários, associação de bibliografia a aulas, etc., são algumas das inúmeras possibilidades. Por si, estes não são novos serviços, nem deverão substituir os existentes. São no entanto serviços e funcionalidades disponibilizados para o utilizador numa área pessoal: a sua montra de serviços, o seu espaço de trabalho.

Este foi, no fundo, um projecto de longo curso, uma verdadeira maratona, com diversos “sprints” pelo meio, tendo a data de 16 de Dezembro de 2006 ficado marcada como um ponto de viragem na lógica de mostrar os sistemas de informação da universidade.

O my.ua serve o seu propósito original: a demonstração da viabilidade de sistemas autónomos, personalizáveis e com integração de informação e serviços de diversas fontes. Nos primeiros seis meses de actividade, havia já cerca de 3500 perfis de utilizador criados e 10 definições de perfis base. Os 3500 perfis representavam cerca de 25% da população universitária activa (www.ua.pt, 2007), o que é ainda assim, um número considerável, pois apesar de não serem obrigados a utilizar o sistema, estes consideraram-no útil. Contudo, o número de perfis base definido é baixo. Tendo em consideração que se trata de um sistema em produção, visitado diariamente por cerca de 10% da população universitária, pode-se considerar que a solução apresentada é válida do ponto de vista de modelação de utilizadores.

Estes 10% poderão aumentar, se aumentar a granularidade do sistema através da definição de perfis base mais específicos e incluindo novos serviços, funcionalidades e componentes.

Apesar de toda a ênfase inicial colocada em torno das questões de SSO, não foi ainda possível, do ponto de vista prático, implementar esta funcionalidade por falta de plataforma institucional que permitisse testar a integração. Contudo, considera-se que esta falha poderá ser colmatada aquando da implementação da plataforma institucional de SSO, visto que o my.ua será “apenas” mais um dos serviços a integrar neste funcionamento.

10 Referências

- Apache Software Foundation. (2007). Lucene.Net. Em linha <http://incubator.apache.org/lucene.net/>. Consultado em 14 Nov. , 2007
- Apache Software Foundation. (2008). Pluto - Welcome to Pluto. Em linha <http://portals.apache.org/pluto/>. Consultado em 16/07, 2008
- Balzer, Y. (2004, 16/07). Improve your SOA project plans. Em linha <http://www-128.ibm.com/developerworks/webservices/library/ws-improvesoa/>. Consultado em 01/10, 2008
- BEA Systems Inc. (2008). BEA - Business Software, Business Process Management, Service Bus, Service Oriented Architecture. Em linha http://www.bea.com/framework.jsp?CNT=homepage_main.jsp&FP=/content. Consultado em 17/07, 2008
- Beged-Dov, G., Brickley, D., Dornfest, R., Davis, I., Dodds, L., Eisenzopf, J., et al. (2001, 30/05). RDF Site Summary (RSS) 1.0. Em linha <http://web.resource.org/rss/1.0/spec>. Consultado em 01/03, 2007
- Bellas, F. (2004). Standards for Second-Generation Portals. *IEEE Internet Computing*, 8(2), 54-60.
- Bergstedt, S., Wiegrefe, S., Wittmann, J., & Moller, D. (2003, 9-11 July 2003). *Content management systems and e-learning systems - a symbiosis?* Artigo apresentado em The 3rd IEEE International Conference on Advanced Learning Technologies, Athens, Greece.
- Billig, A., Gottschick, J., & Sandkuhl, K. (2005, 30/08-3/09). *Evolution of Web computing systems: experiences from Web-portal projects*. Artigo apresentado em 31st EUROMICRO Conference on Software Engineering and Advanced Applications, Porto, Portugal.
- Booth, D., Haas, H., McCabe, F., Newcomer, E., Champion, M., Ferris, C., et al. (2004, 11/02). Web Services Architecture. Em linha <http://www.w3.org/TR/ws-arch/>. Consultado em 21/02, 2005
- Bourret, R. (2004). XML-DBMS: Middleware for Transferring Data between XML Documents and Relational Databases. Em linha <http://www.rpbourret.com/xmldbms/index.htm>. Consultado em 23/11, 2004
- Brad, A. M. (1998). A brief history of human-computer interaction technology. *interactions*, 5(2), 44-54.
- Brajnik, G., & Tasso, C. (1994). A shell for developing non-monotonic user modeling systems. *International Journal of Human-Computer Studies*, 40(1), 31-62.
- C. Tsai, J. H., T. Liang, C. Yang. (2005, 10-12/10). *An Intelligent Web Portal System for Web Information Region Integration*. Artigo apresentado em 2005 IEEE International Conference on Systems, Man and Cybernetics, Hawaii, USA
- C. Zirpins, H. W., A. Bartelt, W. Lamersdor. (2001, 3-7/09/2001). *Advanced Concepts for Next Generation Portals*. Artigo apresentado em 12th International Workshop on Database and Expert Systems Applications, Munich, Germany.

- Carvalho, J. (2005). *An XML representation of the UNIMARC Manual: a working prototype*. Artigo apresentado em World Library and Information Congress: 71th IFLA General Conference and Council.
- Casanova, D., Pinto, J. S., Teixeira, C., & Monteiro, A. (2007, 14-17 May). *The new institutional portal of UA methodology of development*. Artigo apresentado em 2007 Euro American conference on Telematics and Information Systems, Faro, Portugal.
- Chaoying, M., Liz, B., Miltos, P., & Gill, W. (2006, 19-23 Feb.). *Towards the Design of a Portal Framework for Web Services Integration*. Artigo apresentado em Advanced International Conference on Telecommunications / Internet and Web Applications and Services, Guadeloupe, French Caribbean.
- Chappell, D. (2008). Introducing Windows CardSpace. Em linha <http://msdn2.microsoft.com/en-us/library/aa480189.aspx>. Consultado em 02/04, 2008
- Christensen, E., Curbera, F., Meredith, G., & Weerawarana, S. (2001, 15/03). Web Services Description Language (WSDL) 1.1. Em linha <http://www.w3.org/TR/2001/NOTE-wsdl-20010315>. Consultado em 01/10, 2008
- Corporation, O. (2004). The Oracle XML Developer's Kit. Em linha <http://technet.oracle.com/tech/xml/>. Consultado em 05/11, 2004
- Cover, R. (2008). Cover Pages: Security Assertion Markup Language (SAML). Em linha <http://xml.coverpages.org/saml.html>. Consultado em 07/04, 2008
- Covington, M. A., Nute, D., & Vellino, A. (1996). *Prolog Programming in Depth* Prentice Hall.
- Daigle, S. L., & Cuocco, P. M. (2002). Portal Technology Opportunities, Obstacles, and Options - A View from the California State University. In R. N. K. Associates (Ed.), *Web Portals and Higher Education - Technologies to Make IT Personal* (Edição ed., pp. 109-123): Jossey-Bass.
- DeWitt, D. J. (1993). The Wisconsin Benchmark: Past, Present, and Future. In J. Gray (Ed.), *Benchmark Handbook for Database and Transaction Processing Systems* (Edição ed., pp. 269-316): Morgan Kaufmann.
- DuCharme, B. (2004, 11/02). Googling for XML. Em linha <http://www.xml.com/pub/a/2004/02/11/googlexml.html>. Consultado em 03/04, 2005
- E. Bertino, B. C., W. Q. Wang. (2004, March 2004). *XJoin Index: Indexing XML Data for Efficient Handling of Branching Path Expressions*. Artigo apresentado em 20th Inter. Conference on Data Engineering (ICDE'2004), Boston, MA, USA.
- EduRoam. (2008). eduroam. Em linha <http://www.eduroam.org/>. Consultado em 03/04, 2008
- English, B., & Microsoft SharePoint Teams. (2004). SharePoint Portal Server Architecture. In M. Press (Ed.), *Microsoft® SharePoint® Products and Technologies Resource Kit* (Edição ed., pp. 1216): Microsoft.

- Erl, T. (2008). What is SOA. Em linha <http://www.whatissoa.com/>. Consultado em 09/10, 2008
- Finin, T. W. (1989). GUMS: A General User Modeling Shell. In A. Kobsa & W. Wahlster (Eds.), *User Models in Dialog Systems* (Edição ed., pp. 411-430). New York, NY, USA Springer-Verlag New York, Inc.
- G. Adomavicius, A. T. (1999). *User Profiling in Personalization Applications through Rule Discovery and Validation*. Artigo apresentado em 5th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Diego, California, United States
- Gailey, J. H. (2003). *Understanding Web Services Specifications and the WSE* (1 ed.): Microsoft Press.
- Google. (2008). Google Analytics. Em linha <http://www.google.com/analytics/>. Consultado em 03/11, 2008
- Goudner, M., Hondo, M., Nadalin, A., McIntosh, M., & Schimdt, D. (2007, 01/05). Understanding WS-Federation. Em linha http://download.boulder.ibm.com/ibmdl/pub/software/dw/specs/ws-fed/WS-FederationSpec05282007.pdf?S_TACT=105AGX04&S_CMP=LP. Consultado em 01/08, 2008
- Gudgin, M., Hadley, M., Mendelsohn, N., Moreau, J.-J., Nielsen, H. F., Karmarkar, A., et al. (2007, 27/05). SOAP Version 1.2 Part 1: Messaging Framework (Second Edition). Em linha <http://www.w3.org/TR/2007/REC-soap12-part1-20070427/>. Consultado em 22/10, 2008
- Haim, K. (1990). *From semantic to object-oriented data modeling*. Artigo apresentado em First International Conference on Systems Integration '90, Morristown, New Jersey, United States.
- Hallam-Baker, P., Kaler, C., Monzillo, R., & Nadalin, A. (2004, 01/03). Web Services Security X.509 Certificate Token Profile. Em linha <http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-x509-token-profile-1.0.pdf>. Consultado em 01/03, 2007
- IBM. (2008). IBM business collaboration solution - Lotus Domino. Em linha <http://www-01.ibm.com/software/lotus/products/domino/>. Consultado em 06/10, 2008
- IETF - Internet Engineering Task Force. (2008). Public-Key Infrastructure (X.509) (pkix) Charter. Em linha <http://www.ietf.org/html.charters/pkix-charter.html>. Consultado em 17/07, 2008
- IFLA. (2000, 06/04). UNIMARC Manual : Bibliographic Format 1994 - IFLA Universal Bibliographic Control and International MARC Core Programme (UBCIM). Em linha <http://www.ifla.org/VI/3/p1996-1/sec-uni.htm>. Consultado em 11/02, 2008
- Internet2. (2006). Shibboleth Project - Internet2 Middleware. Em linha <http://shibboleth.internet2.edu/>. Consultado em 09/03, 2006
- JA-SIG. (2006). uPortal by JA-SIG. Em linha <http://www.uportal.org/>. Consultado em 01/03, 2006

- Jason, M., Serge, A., Roy, G., Dallas, Q., & Jennifer, W. (1997). Lore: a database management system for semistructured data. *SIGMOD Rec.*, 26(3), 54-66.
- Judy, K., Bob, K., & Piers, L. (2002, May 29-31, 2002). *Personis: A Server for User Models*. Artigo apresentado em Second International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems, Málaga, Spain.
- Jungsuk Song, W. K. (2004, 14-16 Sept. 2004). *Extensible index technique for storing and retrieving XML documents*. Artigo apresentado em The Fourth International Conference on Computer and Information Technology (CIT'04) BRAC University, Bangladesh.
- K. M. Win, W. K. N., E. P. Lim. (2003). *An Architectural Framework for Native XML Data Management*. Artigo apresentado em Second International Conference on Cyberworlds, Singapore.
- Kobsa, A. (1995, July 9-14, 1995). *Supporting user interfaces for all through user modeling*. Artigo apresentado em Sixth International Conference on Human-Computer Interaction, Yokohama, Japan.
- Kobsa, A. (2001). Generic User Modeling Systems. *User Modeling and User-Adapted Interaction*, 11(1-2), 49-63.
- Kobsa, A., & Fink, J. (2006). An LDAP-based User Modeling Server and its Evaluation. *User Modeling and User-Adapted Interaction*, 16(2), 129-169.
- Kobsa, A., Koenemann, J., & Pohl, W. (2001). Personalized Hypermedia Presentation Techniques for Improving Online Customer Relationships. *The Knowledge Engineering Review*, 16(2), 111-155.
- Kroenke, D. M. (2005). Beyond the Relational Database Model. *Computer*, 38(5), 89-90.
- Kropp, A., Leue, C., & Thompson, R. (2003, 09/03). Web Services for Remote Portlets Specification. Em linha <http://www.oasis-open.org/committees/download.php/3343/oasis-200304-wsrp-specification-1.0.pdf>. Consultado em 13/07, 2007
- Letecky, D. (2006). Home | DotLucene (The Fastest Open Source Fulltext Search Engine for .NET). Em linha <http://www.dotlucene.net/>. Consultado em March 2006, 2006
- Liberty Alliance Project. (2008). Home - Liberty Alliance. Em linha <http://www.projectliberty.org/>. Consultado em 07/04, 2008
- Liliana, A., & Anna, G. (2000). Tailoring the Interaction with Users in Web Stores. *User Modeling and User-Adapted Interaction*, 10(4), 251-303.
- Lockhart, H., Andersen, S., Bohren, J., Bohren, J., Hondo, M., Maruyama, H., et al. (2006, 01/12). Web Services Federation Language (WS-Federation) V1.1. Em linha http://download.boulder.ibm.com/ibmdl/pub/software/dw/specs/ws-fed/WS-Federation-V1-1B.pdf?S_TACT=105AGX04&S_CMP=LP. Consultado em 07/05, 2007
- Lucene. (2006). Welcome to Lucene! Em linha <http://lucene.apache.org/>. Consultado em March 2006, 2006

- Mabanza, N., Chadwick, J., & Rao, G. S. V. R. K. (2006). *Performance evaluation of Open Source Native XML databases - A Case Study*. Artigo apresentado em The 8th International Conference on Advanced Communication Technology, Korea.
- Madsen, P., & Maler, E. (2005, 12/04). SAML V2.0 Executive Overview. Em linha <http://www.oasis-open.org/committees/download.php/13525/sstc-saml-exec-overview-2.0-cd-01-2col.pdf>. Consultado em 21/11, 2006
- Margaria, T. (2007). Service is in the eyes of the beholder. *Computer*, 40(11), 124.
- Microsoft Corporation. (2006, 01/04). Introduction to Windows Live ID. Em linha <http://msdn.microsoft.com/en-us/library/bb288408.aspx>. Consultado em 01/08, 2008
- Microsoft Corporation. (2007). Microsoft Office Sharepoint Server. Em linha <http://www.microsoft.com/sharepoint/>. Consultado em 13/07, 2008
- Middleton, S. E., Shadbolt, N. R., & Roure, D. C. d. (2004). Ontological User Profiling in Recommender Systems. *ACM Transactions on Information Systems*, 22(1), 54–88.
- MODIS: ISPRAS. (2008). Sedna XML database. Em linha <http://modis.ispras.ru/sedna/>. Consultado em 11/02, 2008
- Nadalin, A., Goodner, M., Gudgin, M., Barbir, A., & Granqvist, H. (2007a, 01/07). WS-SecurityPolicy 1.2. Em linha <http://docs.oasis-open.org/ws-sx/ws-securitypolicy/200702/ws-securitypolicy-1.2-spec-os.html>. Consultado em 01/09, 2008
- Nadalin, A., Goodner, M., Gudgin, M., Barbir, A., & Granqvist, H. (2007b, 19/03). WS-Trust 1.3. Em linha <http://docs.oasis-open.org/ws-sx/ws-trust/200512/ws-trust-1.3-os.html>. Consultado em 01/09, 2008
- Neuman, B. C., & Ts'o, T. (1994). Kerberos: an authentication service for computer networks. *Communications Magazine, IEEE*, 32(9), 33-38.
- Oasis-open.org (2008). Reference Architecture for Service Oriented Architecture Version 1.0 - Public Review Draft 1. *Journal*. Consultado em <http://docs.oasis-open.org/soa-rm/soa-ra/v1.0/soa-ra.pdf>
- OASIS. (2006, 21/02). OASIS Web Services Security (WSS) TC. Em linha http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wss. Consultado em 01/03, 2007
- OASIS. (2007, 09/05). Web Services Business Process Execution Language Version 2.0. Em linha <http://www.oasis-open.org/committees/download.php/23964/wsbpel-v2.0-primer.htm>. Consultado em 05/09, 2008
- OASIS. (2008). OASIS eXtensible Access Control Markup Language (XACML) TC. Em linha http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=xacml. Consultado em 01/08, 2008
- P. Rao, B. M. (2004, March 2004). *PRIX: Indexing and Querying XML using Prüfer Sequences*. Artigo apresentado em 20th Inter. Conference on Data Engineering (ICDE'2004), Boston, MA, USA.
- Paiva, A., & Self, J. (1994). TAGUS - A user and learner modeling workbench. *User Modeling and User-Adapted Interaction*, 4(3), 197-226.

- Papazoglou, M. P., Traverso, P., Dustdar, S., & Leymann, F. (2007). Service-Oriented Computing: state of the art and research challenges. *Computer*, 40(11), 124.
- Peter, B. (2001). Adaptive Hypermedia. *User Modeling and User-Adapted Interaction*, 11(1-2), 87-110.
- Rich, E. (1979). User modeling via stereotypes. *Cognitive Science*, 3(4), 329-354.
- Rigney, C., Willens, S., Livingston, Rubens, A., Merit, Simpson, W., et al. (2000, 01/06). Remote Authentication Dial In User Service (RADIUS). Em linha <http://www.ietf.org/rfc/rfc2865.txt>. Consultado em 01/08, 2008
- Santos, B. S., Dias, P., Pais, H., Pires, J., Luís, G., Mar, P. d., et al. (2007). Uma avaliação de Usabilidade da aplicação www.abc.ua.pt. *Revista do Departamento de Electrónica, Telecomunicações e Informática da Universidade de Aveiro*, 4(8), 6.
- Santos, B. S., Teixeira, L., Teixeira, B., Lucas, C., Pacheco, D., Ramos, I., et al. (2008). Avaliação de usabilidade de alguns aspectos de um serviço integrado no portal ua.pt: My.UA. *Revista do Departamento de Electrónica, Telecomunicações e Informática da Universidade de Aveiro*, 4(9).
- SAP. (2008). SAP - Business Software Solutions Applications and Services. Em linha <http://www.sap.com/index.epx>. Consultado em 23/10, 2008
- Smith, M. A. (2004). Portals: towards an application framework for interoperability. *Communications of the ACM*, 47(10), 93-97.
- Strauss, H. (2002). All About Web Portals: A Home Page Doth Not a Portal Make. In R. N. K. Associates (Ed.), *Web Portals and Higher Education - Technologies to Make IT Personal* (Edição ed., pp. 32-41): Jossey-Bass.
- Sun microsystems. (2008). Java SE Technologies - Database. Em linha <http://java.sun.com/javase/technologies/database/>. Consultado em 29/10, 2008
- Teixeira, C., Pinto, J. S., & Martins, J. A. (2006, Feb. 2006). *Contact@UA - A Profile Driven Portal*. Artigo apresentado em ICIW 06: International Conference on the Internet and Web Applications and Services, Guadeloupe, France.
- Teixeira, C., Pinto, J. S., & Martins, J. A. (2008a). *Managing scientific production*. Artigo apresentado em Information Technology, 2008. IT 2008. 1st International Conference on.
- Teixeira, C., Pinto, J. S., & Martins, J. A. (2008b, 10-12 Sep.). *UNIMARC-XML Performance Testing*. Artigo apresentado em EATIS: Euro American Conference on Telematics and Information Systems, Aracaju, Brasil.
- Teixeira, C., Pinto, J. S., & Martins, J. A. (2008c, 8-13 Jun.). *User Profiles in Corporate Scenarios*. Artigo apresentado em Third International Conference on Internet and Web Applications and Services, Athens, Greece.
- Teixeira, C., Pinto, J. S., & Santos, J. (2005, 7-9 Sep.). *Contact@UA - Multi Service Web Portal*. Artigo apresentado em ITA 05: First International Conference on Internet Technologies and Applications, Wrexham, North Wales.
- TIBCO. (2008a). Company. Em linha <http://www.tibco.com/company/default.jsp>. Consultado em 03/07, 2008

- TIBCO. (2008b). Portal Software. Em linha <http://www.tibco.com/software/portal/default.jsp>. Consultado em 17/07, 2008
- Tim, F., & David, D. (1986). *GUMS: a general user modeling system*. Artigo apresentado em Proceedings of the workshop on Strategic computing natural language.
- TMN. (2005). myTMN. Em linha <http://www.mytmn.pt/web/mytmn.po>. Consultado em 25/08, 2005
- Vignette. (2008). Vignette - Enterprise Content Management and Portal Solutions. Em linha <http://www.vignette.com/>. Consultado em 20/07, 2008
- VMware. (2008). VMware ESX, Bare-Metal Hypervisor for Virtual Machines - VMware. Em linha <http://www.vmware.com/products/vi/esx/>. Consultado em 22/07, 2008
- Vodafone. (2005). main. Em linha <https://my.vodafone.pt>. Consultado em 25/08, 2005
- W. Zhang, D. L., J. Li. (2004). *An Encoding Scheme for Indexing XML Data*. Artigo apresentado em 2004 IEEE International Conference on e-Technology, e-Commerce and e-Service (EEE'04). de <http://doi.ieeecomputersociety.org/10.1109/EEE.2004.1287357>
- Williams, S. (2008). What is a Content Management System, or CMS? Em linha <http://www.contentmanager.eu.com/history.htm>. Consultado em 18/10, 2008
- Wugofski, T. (2000). *XML Black Book 2nd Edition: The Complete Reference for XML Designers and Content Developers* (2 ed.): Coriolis Group Books.
- www.ua.pt. (2007). Universidade de Aveiro : factos e números. Em linha <http://www.ua.pt/PageText.aspx?id=429> Consultado em 08/06, 2007
- Zhang, J., & Ghorbani, A. A. (2007, 14-17 May). *GUMSAWS: A Generic User Modeling Server for Adaptive Web Systems*. Artigo apresentado em Fifth Annual Conference on Communication Networks and Services Research, Frederlcton, New Brunswick, Canada.
- Zimmermann, A., Specht, M., & Lorenz, A. (2005). Personalization and Context Management. *User Modeling and User-Adapted Interaction*, 15(3-4), 275-302.
- Zou, J., & Pavlovski, C. J. (2007, 24-26 Oct.). *Towards Accountable Enterprise Mashup Services*. Artigo apresentado em IEEE International Conference on e-Business Engineering, Hong Kong.