

UNIVERSIDADE DE LISBOA
FACULDADE DE CIÊNCIAS
Departamento de Informática



PORTAL DE GESTÃO DO UTILIZADOR DA ULISBOA

Ricardo Miguel Rodrigues Rito

Mestrado em Engenharia Informática
Especialização em Sistemas de Informação

Dissertação orientada por:
Professor João Diogo Silva Ferreira e Dra. Ana Rute Marques

[2017]

“My model for business is The Beatles. They were four guys who kept each other’s kind of negative tendencies in check. They balanced each other and the total was greater than the sum of the parts. That’s how I see business: great things in business are never done by one person, they’re done by a team.”

STEVE JOBS

Agradecimentos

Necessitaria de mais algumas páginas para agradecer a todos aqueles que contribuíram para que esta tese de Mestrado em Engenharia Informática chegasse a bom porto. Foram muitas as pessoas que me ajudaram, direta ou indiretamente, a cumprir os meus objetivos, que culminam em mais uma etapa da minha formação académica. Vou deixar algumas palavras de agradecimento, e espero não me esquecer de ninguém...

Ao Mark Knopfler, Jeff Beck, Sting, Beatles, Joe Bonamassa, John Mayer, Rui Veloso, José Cid, Bob Dylan, Robben Ford, Artur Emídio, Paulo Saraiva e Bruno Macedo, por toda a música que criaram. Foram uma companhia musical assídua durante as pesquisas, trabalho e redação desta dissertação.

A todos aqueles que externamente permitiram para que este projeto se concretizasse, nomeadamente à Sandrinha do Senhor António, ao Nuno Marques, ao Nicolas e ao Rui Vidigal pela clareza e total disponibilidade, para a colaboração no tratamento dos dados estatísticos, formatações e principalmente pela contagiante boa disposição.

Aos colegas do Núcleo de Comunicação, nomeadamente ao Marco e à Sofia, pela dedicação com que encararam o desafio da elaboração da imagem do Portal do Utilizador, e a transformou em algo de extraordinário.

A todos os colegas do Departamento de Informática, nomeadamente aos colegas dirigentes e superiores, Professor Carlos Ribeiro, Dr. Sérgio Vicente, Eng. André Brioso, Dr. Nuno Abrantes e Dra. Célia Nunes.

A todos os colegas do Núcleo de Sistemas e Infraestruturas, do Núcleo de Administração de Redes e Telecomunicações e da Área de Suporte e do departamento de Informática pela paciência que tiveram nos meus pedidos.

A todos os colegas do Núcleo de Gestão de Sistemas de Informação, pela gestão do projeto, em especial ao José Pedro, Rui Manuel, Diogo Gustavo, Diogo Alexandre, José Eusébio, Andreia Sofia e à Daniela Luísa.

Aos meus queridos “minions” da equipa do Núcleo de Desenvolvimento de Software: Ana Catarina, Eduardo, Pedro Manuel e Nuno Miguel, por todo o apoio, espírito de entreajuda, boa disposição, disponibilidade, e pela capacidade de me fazer sorrir. Um bem-haja. Sou feliz com vocês :). Obrigado!

Ao João Manuel, pela paciência, capacidade expressiva, comunicação, impulsão, disponibilidade, alerta nas divergências, opiniões críticas e por todo o apoio. Não tenho palavras para agradecer. É bom ter amigos assim!

À minha coorientadora Dra. Ana Rute Marques, que para além de grande impulsionadora deste desafio, foi sempre uma pessoa que me permitiu sonhar, para me deixar crescer. A visão macro deste projeto foi elaborada pela Ana. Sem a Ana, o desfecho deste projeto seria certamente diferente.

Ao meu orientador Professor Doutor João D. Ferreira, pela serenidade, orientação, capacidade de análise, partilha do saber e valiosas contribuições para o trabalho. Um bem-haja por me acompanhar neste desafio e por estimular o interesse pelo conhecimento.

Dedico esta tese aos meus pais, à Lucinda, ao Amável, às minhas queridas filhas Sarah Graça e Maria, bem como à minha incansável Carine por todo o suporte, carinho e estímulo.

Resumo

O conceito de identidade na ULisboa (Universidade de Lisboa) segue uma abordagem típica dos sistemas de IdM (Identity Management) em que qualquer indivíduo (aluno, ex-aluno, funcionário docente, não-docente ou outro) é único, e a sua informação e registo é gerido como tal, com recurso a perfilagem (uma entidade pode pertencer a uma ou a várias perfilagens, sendo a perfilagem o conceito que distingue o tipo de acesso a um determinado sistema).

A gestão da informação de identidade de cada indivíduo tem origem nos processos das áreas Académica, para alunos, e de Recursos Humanos, para funcionários. Para cada indivíduo existem sistemas, procedimentos, regras e informação específica que determinam o estado da sua identidade e perfil.

Foi objetivo desta tese desenvolver um sistema denominado de Portal do Utilizador (PU), que irá providenciar aos utilizadores um acesso fácil e direto respeitante à sua identidade, baseado na web. Neste documento é descrito o processo de desenvolvimento do PU implementado por mim no Departamento de Informática dos Serviços Centrais da Universidade de Lisboa. O PU estará disponível em todas as escolas da ULisboa, para alunos, ex-alunos, funcionários docentes e funcionários não docentes, e tem como objetivo centralizar as várias funcionalidades relativas à criação e gestão de identidade na ULisboa, em integração direta com o IdM da ULisboa. Estas funcionalidades incluem: criação e ativação das contas dos utilizadores da ULisboa, criação de utilizadores temporários, edição do perfil de utilizador (alteração de senha de acesso e de e-mail externo, entre outras), ativação de conta para acesso aos serviços Google e Office 365 (para utilizadores que pertencem a uma escola que não utiliza a conta única da ULisboa como conta institucional) e acesso a dados pessoais e histórico.

O desenvolvimento do PU teve por base requisitos funcionais e técnicos indicados pelos Serviços Centrais da ULisboa, fundamentados nas respostas a um questionário de utilizadores, e assenta em tecnologias de engenharia de aplicações web como RichFaces, JSF, Java EE, Hibernate, JBoss, jQuery e Bootstrap.

Palavras-chave: IdM, Identity Manager, LDAP, Portal da ULisboa

Abstract

The concept of identity in ULisboa (University of Lisbon) follows a standard Identity Management (IdM) based approach. Hence, each user is unique, and at a functional level, in the processes that manages the information and their dependencies and at a technological level, in the infrastructure that stores and manages that information.

The Identity Management of each user has different sources: student profile origin's is Academic System Management based, employee profile is HR System based. Each individual profile is composed by several specifications – based in rules, procedures and systems – that establish the user's identity status and profile. These two areas have systems, procedures, rules and specific managed information that results on a user identity and a profile. There are also a set of processes and systems out the manage core, that depends on that information to grant access privileges and verify rules.

Each ULisboa Organic Unit manages their information in a different way with specific rules and procedures.

The objective of this thesis is to develop a system called 'Portal do Utilizador' and deploy it on the Central Services of ULisboa, making it available to every Organic Unit, (ex-)student and employees.

The “Portal de Utilizador” development was built respecting some functional and technical requisites required by the ULisboa's Central Services. Portal do Utilizador will be responsible to manage ULisboa accounts with direct integration of ULisboa IdM.

Keywords: IdM, eDirectory, LDAP, ULisboa

Conteúdo

Capítulo 1	Introdução	1
1.1	Motivação	1
1.2	Objetivos	2
1.3	Onde o trabalho foi realizado	2
1.4	Resultados obtidos	3
1.5	Organização do documento	3
Capítulo 2	Trabalho relacionado – Estado da arte	5
2.1	O que é a gestão de Identidade?	6
2.2	LDAP	8
2.3	IdM	11
2.3.1	eDirectory	12
2.3.2	iManager	13
2.4	O IdM na ULisboa	13
2.4.1	Perfilagem	16
2.4.2	Schema e Atributos	18
2.4.3	Regras de <i>Matching</i>	19
2.4.4	Regras de Aprovisionamento e Desaprovisionamento	20
2.4.5	Conectores IdM	24
2.4.6	O portal do utilizador no IdM	28
2.5	Modelo Sequencial Linear	28
Capítulo 3	Tecnologias Utilizadas no Portal do Utilizador	32
3.1	Arquitetura	34
3.2	Ferramentas	35
3.2.1	XHTML	35
3.2.2	JavaScript	35
3.2.3	RichFaces	37
3.2.4	Bootstrap	37
3.2.5	JSF	38
3.2.6	Java EE	38
3.2.7	LDAP Novell	39
3.2.8	Hibernate	39
3.2.9	Sistema de gestão de bases de dados Oracle	40

3.2.10	JBoss Seam	40
3.2.11	JBoss Server.....	42
3.2.12	TestNG.....	42
3.2.13	GIT	42
Capítulo 4	O trabalho.....	44
4.1	Análise de Utilizadores e Tarefas	45
4.1.1	Análise Estatística dos resultados	46
4.1.2	Interpretação dos resultados.....	56
4.1.3	Análise de tarefas	58
4.2	Desenho técnico e funcional	63
4.3	Implementação.....	67
4.4	Testes.....	68
4.5	Manutenção	68
Capítulo 5	Discussão e conclusão	70
	Referências Bibliográficas	72
	Glossário de Acrónimos	76
Anexo I	Valores dos sufixos de atributos	i
Anexo II	Atributos da classe ULAuxUser	ii
Anexo III	Atributos da classe ULAuxFac[...].	iv
Anexo IV	Questionário	vii
Anexo V	Regras das senhas de acesso.....	xii
Anexo VI	Descrição dos casos de uso	xiii
Anexo VII	Protótipos de baixa fidelidade	xix

Capítulo 1

Introdução

Pretende-se neste primeiro capítulo, efetuar uma breve contextualização e apresentar a motivação para o trabalho.

Neste capítulo também serão expostos os objetivos a que me propus neste projeto, e apresentarei também a organização deste documento.

1.1 Motivação

Pretende-se substituir o atual sistema de gestão de identidades gerido pelos Serviços Centrais da Universidade de Lisboa (ULisboa), por um novo sistema de gestão de identidades.

O objetivo do projeto de gestão de Identidade na ULisboa consiste na instalação, implementação e desenvolvimento de um sistema de meta diretório com integração em vários sistemas na ULisboa, sistemas estes que são responsáveis por funcionalidades como a autenticação, a autorização de acessos, os sistemas de bases de dados para persistência da informação, e a replicação (entre estes serviços encontram-se entre os quais “Active Directory” (Microsoft), Bases de dados Oracle, e sistemas de replicação em “LDAP” (*Lightweight Directory Access Protocol*), na ULisboa.

É necessário que neste projeto sejam asseguradas todas as configurações necessárias para o aprovisionamento dos dados e funcionamento dos serviços e aplicações que atualmente dependem do sistema de gestão de identidades central.

Para atingir o objetivo proposto recorreu-se a *software* Novell - IdM, fazendo uso do motor do meta diretório LDAP *eDirectory*.

É utilizado também o *software iManager* que funciona como uma interface administrativa web entre o administrador de sistemas e o diretório LDAP. É no *iManager* que são configurados os conectores de ligação aos sistemas externos.

Um dos componentes que faz parte deste sistema de gestão de identidades é uma interface que permita aos utilizadores – neste caso os indivíduos que fazem parte do universo da ULisboa, ou seja os seus funcionários, docentes, alunos e ex-alunos – acederem à informação que o sistema lhes associa. Assim, uma das tarefas na correta implementação deste sistema é a implementação dessa interface, nomeadamente sob a forma de uma aplicação web que seja acessível a todos os utilizadores da ULisboa.

1.2 Objetivos

O projeto onde fui inserido tem como objetivo a implementação do Portal de Utilizador, substituindo a atual interface web de ativação, alteração da senha de acesso e registo de utilizadores que existia na ULisboa.

Este novo Portal de Utilizador (PU), visa proporcionar aos utilizadores da ULisboa, uma interface para:

- Ativação de conta com registo de utilizador;
- Alteração de senha de acesso;
- Recuperação da senha de acesso;
- Criação de utilizadores temporários;
- Alteração do e-mail externo para recuperação da senha de acesso;
- Escolha do domínio de e-mail principal;
- Criação de conta *Google for Education* e *Office 365* (para utilizadores – alunos, docentes e não docentes – das Escolas da ULisboa que não estão ao abrigo do projeto);
- Acesso a dados pessoais e histórico

Esta componente irá estabelecer trocas de informação com *eDirectory* através da escrita e leitura de atributos LDAP que vão despoletar os eventos nos conectores do IdM (iManager) e propagação do pedido.

1.3 Onde o trabalho foi realizado

Após ter concluído a Licenciatura em Engenharia Informática na Faculdade de Ciências da Universidade de Lisboa em 2009, tomei a decisão de efetuar a minha inscrição no Mestrado em Engenharia Informática na mesma Universidade. Após a conclusão das disciplinas da parte curricular e sendo a minha especialização em

Sistemas de Informação, decidi candidatar-me ao estágio “Portal de Gestão de Utilizador Da ULisboa”, sobre a orientação do Prof. João Diogo Silva Ferreira, coorientado pela Dra. Ana Rute Ferreira, Coordenadora da Área de Sistemas de Informação do DI – Serviços Centrais – Reitoria da Universidade de Lisboa.

O meu projeto foi realizado no Departamento de Informática da Reitoria da Universidade de Lisboa, na Área de Sistemas de Informação, no Núcleo de Desenvolvimento de Software.

1.4 Resultados obtidos

O Portal do Utilizador foi desenvolvido com base num inquérito *on-line* feito aos futuros utilizadores. O novo portal será utilizado pela comunidade académica da ULisboa, funcionários docentes e não docentes e Alumni. Neste momento existem aproximadamente 75 000 utilizadores que irão interagir com o Portal do Utilizador.

O PU foi implementado com base nos requisitos levantados, recorrendo a tecnologias web. Para os testes de aceitação, foi elaborada uma bateria de testes, os quais foram bem-sucedidos.

Foi também aceite uma comunicação para o INForum 2017¹ que se realizará nos dias 12 e 13 de Outubro de 2017, onde será feita uma apresentação do Portal do Utilizador e a apresentação de um poster sobre o projeto.

O Portal do Utilizador estará disponível em: <https://utilizadores.campus.ulisboa.pt>

1.5 Organização do documento

Este documento está organizado da seguinte forma:

Capítulo 1 – Introdução

Neste capítulo é apresentada a motivação, os objetivos desta tese e a sua organização

Capítulo 2 – Trabalho relacionado e estado da Arte

¹ <http://inforum.org.pt/INForum2017>

Conceitos, aplicações e ferramentas que compõem o ecossistema e fazem a ligação entre os vários sistemas.

Capítulo 3 – Tecnologias Utilizadas

Tecnologias utilizadas na implementação do Portal do Utilizador.

Capítulo 4 – O Trabalho realizado

Todo o trabalho subjacente à implementação do Portal de Utilizador, incluindo a análise de utilizadores e tarefas, a prototipagem, a implementação e os testes.

Capítulo 5 – Discussão

Neste capítulo apresentam-se os resultados do projeto implementado e apontam-se conclusões.

Referências Bibliográficas

Referências bibliográficas consultadas para a elaboração deste documento.

Glossário de abreviaturas – Lista com a correspondência dos termos abreviados no documento

Anexos:

Anexo I - Valores dos sufixos de atributos

Anexo II - Atributos da classe ULAuxUser

Anexo III - Atributos das classes ULAuxFac[...]

Anexo IV - Questionário

Anexo V - Regras das senhas de acesso

Anexo VI - Descrição dos casos de uso

Anexo VII - Protótipos de baixa fidelidade

Capítulo 2

Trabalho relacionado – Estado da arte

A implementação do Portal de Utilizador (PU), insere-se no âmbito de um projeto de gestão de Identidade da Universidade de Lisboa (ULisboa). Este projeto é um componente essencial de criação, recuperação, definição e gestão de acessos dos utilizadores da ULisboa.

A atual interface web de registo, criação e alteração da senha de acesso dos utilizadores da ULisboa apresenta algumas lacunas: interface pouco amigável, fraca usabilidade em dispositivos móveis, necessidade de novas funcionalidades solicitadas pelos utilizadores, pelas Escolas e principalmente pela evolução do projeto.

Existia uma forte vontade por parte da Área de Sistemas de Informação do Departamento de Informática dos Serviços Centrais da Reitoria da Universidade de Lisboa (AASI/DI/CS/ULisboa) de substituir o atual site por um Portal de Utilizador mais atual, completo e com mais funcionalidades. Essa vontade nunca se concretizou, pois requeria esforço de desenvolvimento da equipa do NDS que se encontrava alocada a outros projetos também importantes, como por exemplo o FenixEdu, SAP e desenvolvimentos internos.

Estando a decorrer um *upgrade* ao sistema IdM instalado, alterações de arquitetura ao nível dos conectores na ULisboa e a implementação do *Access Manager* na ULisboa, tomou-se a oportunidade e a necessidade para se avançar com este tão desejado Portal de Utilizador.

Qualquer utilizador, sendo ele aluno, funcionário docente ou não docente da ULisboa, necessita deste portal para gerir a sua conta.

O novo portal irá interagir com o sistema de gestão de identidade instalado na ULisboa: o IdM.

Apesar de o meu trabalho de mestrado estar diretamente focado no desenho e implementação do PU, é essencial tanto para esta implementação como para a compreensão do trabalho desenvolvido, que seja descrito com algum pormenor o funcionamento da plataforma que gere a identidade dos vários utilizadores da ULisboa. Assim, este capítulo descreve o IdM (*software de identity management* usado pela ULisboa), focando-se nos seus componentes, nos serviços a ele subjacentes e nas funcionalidades que ele permite executar.

2.1 O que é a gestão de Identidade?

Para os efeitos deste trabalho, convém salientar uma diferença conceptual entre os termos “entidade” e “identidade”. Ao passo que um indivíduo é uma entidade por si mesmo, a sua identidade pode variar de acordo com o contexto (é um aluno na universidade, um cliente numa loja, um contribuinte no contexto fiscal, ...). Assim, uma entidade pode ter várias identidades. Da mesma forma, uma identidade pode corresponder a várias entidades, nomeadamente no caso das organizações, que têm uma identidade coletiva única, à medida que, simultaneamente, são constituídas por vários indivíduos singulares (Bertino & Takahashi, 2011).

A definição de identidade digital no contexto da gestão de identidade não é única. Existem diferentes definições. Há quem defina que a identidade de uma pessoa individual pode compreender muitas identidades parciais das quais cada uma representa a pessoa num contexto ou função específica (Pfitzmann & Hansen, 2010). Uma identidade parcial é um subconjunto de valores de atributos de uma identidade completa, onde uma identidade completa é a união de todos os valores de atributos de todas as identidades dessa pessoa. Uma entidade tem a ver com a integração num grupo social e as entidades parciais, por exemplo, com relações de membros de um grupo em particular – relações com subconjuntos dos membros do grupo. Por outro lado, identidades parciais podem estar associadas a relações com organizações. Nesta definição as pessoas são consideradas como sujeitos de identidades, e não como entidades humanas (Bertino & Takahashi, 2011).

Existem definições de identidade que abrangem uma vasta gama de assuntos, não só pessoas. Revelam que os indivíduos das identidades podem ser agentes de *software* (por exemplo, serviços web e *software* cliente de utilizador) e dispositivos de *hardware* (por exemplo, computadores, dispositivos móveis e equipamento de rede) (Bishop,

2002). Além disso, à medida que os ambientes de computação vão estando presentes em todo o lado, as identidades são atribuídas a objetos artificiais (por exemplo, bens diários, peças de máquinas e edifícios) e objetos naturais (por exemplo, gado e colheitas) monitorizados e controlados por sensores.

Há recomendações de padronizações de conceitos de gestão de identidade que consideram a informação existente numa entidade suficiente para identificar essa identidade num contexto particular (ITU, 2009). Estas recomendações definem a informação associada a uma entidade em três tipos: identificadores, credenciais e atributos. A gestão de informações de uma identidade baseando-se em identificadores (por exemplo, o endereço de correio eletrónico, telefone, identificador único de utilizador), credenciais (certificados digitais, dados biométricos, *tokens*) e atributos (por exemplo, regras de negócio, privilégios de acesso, localização) já recebeu outras abordagens. Cada vez mais os serviços são baseados em contextos e regras, são acedidos a partir de qualquer lugar, a qualquer momento, tornando a gestão de informação contida na identidade, vulnerável a questões de segurança. Existem também desafios como a interoperabilidade dos sistemas em contextos heterogêneos.

A identidade pode abranger uma gama de disciplinas, incluindo a sociologia, psicologia, filosofia, bem como a ciência da computação. Um estudo sobre identidade em ciência da computação revelou que as entidades são categorizadas a partir de perspectivas estruturais e de processo (Nabeth, 2009). De uma perspectiva estrutural, uma identidade é vista como uma representação ou um conjunto de atributos que caracterizam a pessoa. Do ponto de vista do processo, a identidade é formada através da identificação de um conjunto de processos relacionados à divulgação de informações sobre a pessoa e ao uso dessa informação.

Podem ser terceiros a definir uma identidade. A identidade é gerada com base em quem tem e controla as identidades. Este conceito de abordagem de identidade é composta por três níveis: a *minha identidade*, a *nossa identidade* e a *identidade por terceiros* (Searls, 2002). A *minha identidade*, é como um identificador central. Pode ser exposto e anónimo, e é a pessoa própria que controla o acesso dos outros. É também considerada a verdadeira identidade e são criadas quando o objeto pessoa natural nasce. A *nossa identidade* não pertence à pessoa nem à identidade que nos fornece. Esta identidade existe por meio de acordos de terceiros (por exemplo, a identidade como funcionário de uma instituição e que se mantêm enquanto existir um acordo entre a

instituição e o funcionário). A *identidade por terceiros* é uma identidade que é criada internamente, normalmente por uma empresa com interesses comerciais identificados, sem o consentimento explícito do sujeito. Neste tipo de identidade, as empresas criam conjunturas de identidades dos utilizadores, mesmo com a pouca informação que têm do utilizador, utilizando dados com o IP, localização demográfica e *cookies* do computador (por exemplo, um serviço de pesquisa web, que cria um modelo de utilizador com base nas pesquisas para aquele IP).

A identidade de um sujeito não é singular. Um exemplo são as pessoas que têm uma identidade numa determinada rede social, e uma outra como estudante, funcionário ou membro de uma academia de fitness (Bertino & Takahashi, 2011). A sua identidade como utilizador de uma rede social consiste numa conta de utilizador com um identificador, conhecimento da senha de acesso e um nome de utilizador. Tem também agregadas à sua identidade listas de amigos e registos de atividades como atributos. A identidade como empregado consiste num identificador que é um número único, um cartão de identificação com credenciais, cargos, afiliação, localização do escritório e avaliação de desempenho como atributos. Existem características que são partilhadas pelas duas identidades. As identidades de um sujeito podem ser relacionadas (federadas) entre si.

Perante todas as definições de identidade apresentadas e dos tipos de identidade que circulam no quotidiano, o objetivo traçado para a gestão de identidade é a de manter a integridade das identidades através dos seus ciclos de vida, a fim de tornar os dados relacionados disponíveis para os serviços de forma segura e protegida pela privacidade. Um sistema de gestão de identidade tem de ter capacidade para adicionar ou remover o aprovisionamento ou não aprovisionamento da informação. Para isso, deverá assegurar políticas de autenticação e políticas de acesso à informação, sistemas, dados e funcionalidades.

2.2 LDAP

Um serviço de diretório é um repositório central que armazena e fornece acesso a informações de utilizadores, sistemas, redes, serviços e aplicações de rede que necessitam de ser partilhadas através da rede (Oracle Help Center, 2016). Os serviços de diretório podem fornecer qualquer conjunto de registos organizado que obedece geralmente a uma estrutura hierárquica (por exemplo, uma lista telefónica que é uma

lista de assinantes com um endereço e um número de telefone). As grandes empresas utilizam serviços de diretórios para armazenar as informações de funcionários como o nome, departamento, número de segurança social e informações organizacionais.

Um serviço de diretório exige um armazenamento de dados robusto, rápido e altamente escalável para as informações nelas armazenadas. Como resultado, esses serviços normalmente são compilados numa base de dados corporativa. Um serviço de diretório é, portanto, essencialmente uma base de dados especializada com uma organização hierárquica que pode ser facilmente estendido para armazenar uma variedade de diferentes tipos de informações.

O *Lightweight Directory Access Protocol* (LDAP) é um protocolo de aplicação TCP/IP de acesso e manutenção de serviços de diretórios distribuídos que atuam de acordo com os modelos de dados e serviços X.500 (Zeilenga, 2006). O modelo de informações do LDAP (para dados e *namespaces*) é semelhante ao do serviço de diretório X.500 OSI, com algumas limitações funcionais e menores recursos que o X.500 (Microsoft, 2016).

Ao contrário da maioria dos protocolos da Internet, o LDAP possui uma API associada que simplifica o desenvolvimento, acesso e manutenção de serviços de diretório. A API LDAP é aplicável à gestão dos diretórios e aos serviços de pesquisa que não têm suporte de serviço de diretório como função principal. O LDAP não pode criar diretórios nem especificar como é que um serviço opera sobre esse diretório.

O esquema da *Imagem 1* ilustra os vários componentes de um serviço de diretório.

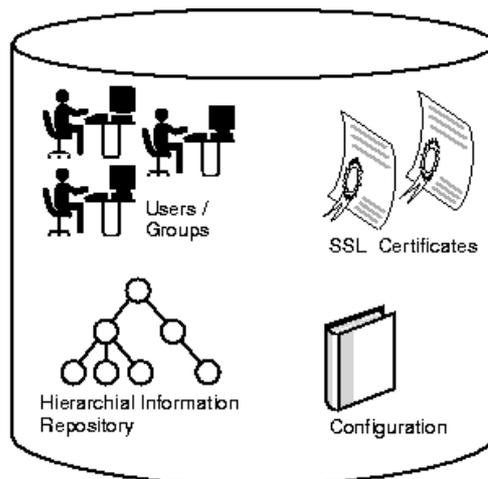


Imagem 1 - Serviço de Diretório LDAP (Oracle Help Center, 2016)

O LDAP define quatro modelos básicos que descrevem as operações, que tipos de informações podem ser armazenados, e o que pode ser feito com as informações (Howes, Smith, & G, 2003). Estes quatro modelos são:

- *Information Model*: define que tipos de informações podem ser armazenados num diretório LDAP. É centrado no conceito de entrada (*entry*). As entradas são criadas para armazenar informações sobre objetos e são compostas por atributos que podem ter um ou mais valores.
- *Naming Model*: define como as informações num diretório LDAP podem ser organizadas e referenciadas. Apesar de não ser um requisito de protocolo, as entradas de diretório LDAP são normalmente organizadas numa estrutura de árvore hierárquica. As entradas são nomeadas de acordo com sua posição na hierarquia por um nome distinto (*distinct name* – DN). Cada componente do DN é chamado um nome distinto relativo (*relative distinct name* – RDN) e é composto por um ou mais atributos da entrada.
- *Functional Model*: define o que fazer com as informações num diretório LDAP e como ele pode ser acessado e atualizado.
- *Security Model*: define como proteger as informações num diretório LDAP e que tipos de privilégios os utilizadores e aplicações necessitam para aceder ao diretório.

A especificação do LDAP está espelhada em inúmeras publicações de sequência de padronização do *Internet Engineering Task Force* (IETF) chamadas *Request for Comments* (RFCs), onde é usada a linguagem de descrição ASN.1.

O LDAP é usado principalmente por organizações de dimensão média ou superior, e tem como principal finalidade a capacidade de partilha de informações de utilizadores, sistemas, redes e serviços através da rede (Software, Gracion, 2016).

A procura de soluções que respondessem a questões diversas, como a pesquisa de endereços de e-mail e a manutenção de uma lista de contactos centralizada e atualizada onde pudesse existir um acesso generalizado, criaram a necessidade em grandes empresas de suportar o padrão LDAP. Através de programas cliente compatíveis com LDAP, consegue-se assim efetuar pedidos ao servidor para que procure entradas de uma variedade de formas diferentes. Os servidores LDAP indexam os dados nas suas entradas e com a utilização de filtros de pesquisa, conseguem a extração de informação

desejada, permitindo a seleção de apenas uma pessoa ou grupo com informações dos atributos desejados.

Por ser um protocolo, um diretório LDAP não define como os programas funcionam do lado do cliente ou do servidor, mas sim a forma como os programas cliente e servidor comunicam entre si. São exemplos os programas de serviços de e-mail e calendário como o Microsoft Outlook.

Um LDAP não contém apenas informações de contactos, ou informações de pessoas. O LDAP permite também disponibilizar informações de uma variedade de serviços, e fornecer o “login único”, onde uma senha de utilizador é partilhada com outros serviços. Outra característica do LDAP é a definição de permissões para restrição de acessos e manutenção de dados privados.

Apesar de o LDAP não integrar um catálogo de endereços de e-mail de todo o mundo (ideia que foi rapidamente inutilizada com o aparecimento de *spam*), continua a ser um padrão popular para a comunicação de dados baseados em registos e de diretório entre os programas.

A versão atual do LDAP é LDAPv3 (History of LDAP, 2017). A especificação teve origem na Universidade de Michigan e foi adotado por inúmeras empresas entre as quais a Microsoft, a Novell, a Netscape, a Lotus e a IBM (Briggs & Spence, 2016).

2.3 IdM

O grande desafio quando se pretende implementar um sistema de gestão de identidade, é o de permitir que “Indivíduos certos acedam aos recursos certos nos momentos certos e pelas razões certas” (Gartner, 2016).

A escolha da solução, após concurso público, para motor de gestão de identidade da ULisboa, foi a da empresa da NetIQ *Identity Manager* IdM², que inclui o iManager e o eDirectory (inicialmente disponibilizada pela Novell, neste momento é uma solução da NetIQ1).

² Página da NetIQ <https://www.netiq.com>

2.3.1 eDirectory

O NetIQ eDirectory é um diretório LDAP que fornece uma grande escalabilidade, sobre uma plataforma ágil e que permite implementar e disponibilizar uma infraestrutura de identidade de uma organização e serviços de rede multiplataforma.

O eDirectory é uma extensão do LDAP e permite o acesso através dos protocolos mais comuns, facilitando a integração com outras soluções existentes LDAP.

O eDirectory organiza os objetos numa estrutura em árvore, começando por um objeto nó principal que indica o nome da árvore. Com esta organização, não é necessário aceder a diferentes domínios ou servidores para criar objetos, gerir permissões, alterar valores de atributos, ou gerir aplicações.

Existem três classes de objetos comuns (ver *Imagem 2*):

- A raiz da árvore (o nó principal). Por norma, contém o objeto de organização
- O objeto Organização é normalmente a primeira classe de repositório de objetos por baixo do objeto árvore.
- Os objetos Unidade Organizacional podem ser criados sob a organização para representar regiões geográficas, campus de rede, ou departamentos individuais. É também possível criar subunidades organizacionais para subdividir mais a árvore.

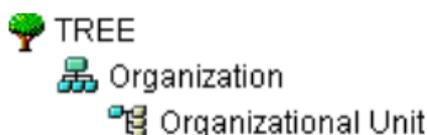


Imagem 2- Classes comuns dos objetos do repositório (adaptado de ³)

Existe um fluxo de permissões, ou seja, existe herança de direitos e atributos, de todas as folhas filhas da árvore.

A definição para cada tipo de objeto do eDirectory é chamada “classe” do objeto. As classes Utilizador e Organização têm certas propriedades que permitem gerir estes tipos de objeto por exemplo, um objeto utilizador, por exemplo, tem uma senha de acesso, apelido, nome para autenticar e outros atributos pré-definidos.

³ Documentação Novell sobre iManager <https://www.novell.com/>

Um *schema* é um conjunto de regras que define as classes e os atributos permitidos num diretório. É no *schema* que é definida a estrutura do diretório, e são apresentadas as relações que as classes têm umas com as outras.

Existem situações em que os *schemas* do diretório LDAP e do diretório do eDirectory são diferentes. Neste caso, pode existir a necessidade de criar um mapeamento de classes e de atributos LDAP para os objetos e atributos do eDirectory apropriados. Estes mapeamentos definem a conversão de nome do *schema* LDAP para o *schema* do eDirectory.

Os serviços LDAP para eDirectory fornecem mapeamentos padrão. Em muitas situações a correspondência entre as classes e atributos LDAP, e dos tipos e propriedades dos objetos eDirectory é lógica e intuitiva. Na maioria dos casos, o mapeamento de tipo de objeto LDAP para o eDirectory é um relacionamento de um para um.

O protocolo LDAP3 permite que clientes LDAP e servidores LDAP usem funcionalidades para estender operações LDAP. Desta forma é possível especificar funções adicionais como parte de um pedido ou de uma resposta.

2.3.2 iManager

O iManager é uma consola de administração web que permite um acesso seguro e personalizado a funcionalidades de administração, gestão e configuração de objetos eDirectory. Este sistema tem a capacidade de atribuição de tarefas e responsabilidades específicas aos utilizadores, disponibilizando as ferramentas necessárias. É um exemplo, a delegação de tarefas comuns como a redefinição de senhas de acesso aos serviços a um grupo de utilizadores, adição/alteração de regras e políticas de conectores, modificação do *schema* ao nível de classes e atributos, a gestão da replicação entre os vários servidores eDirectory, a criação e configuração de modelos para envio de e-mails, entre outras opções.

2.4 O IdM na ULisboa

O IdM é um *software* de gestão de identidade que permite gerir o “ciclo de vida” de um utilizador nos diversos sistemas de informação de uma organização, desde a sua contratação, a alterações consequentes, até ao seu desaproveitamento de todos os sistemas. Ou seja, permite organizar numa única plataforma todos os utilizadores dos

diferentes sistemas de uma organização. A versão que se encontra instalada na Universidade de Lisboa é a *Identity Manager 4.0.2*. Brevemente irá ser feito o *upgrade* para a versão 4.5.

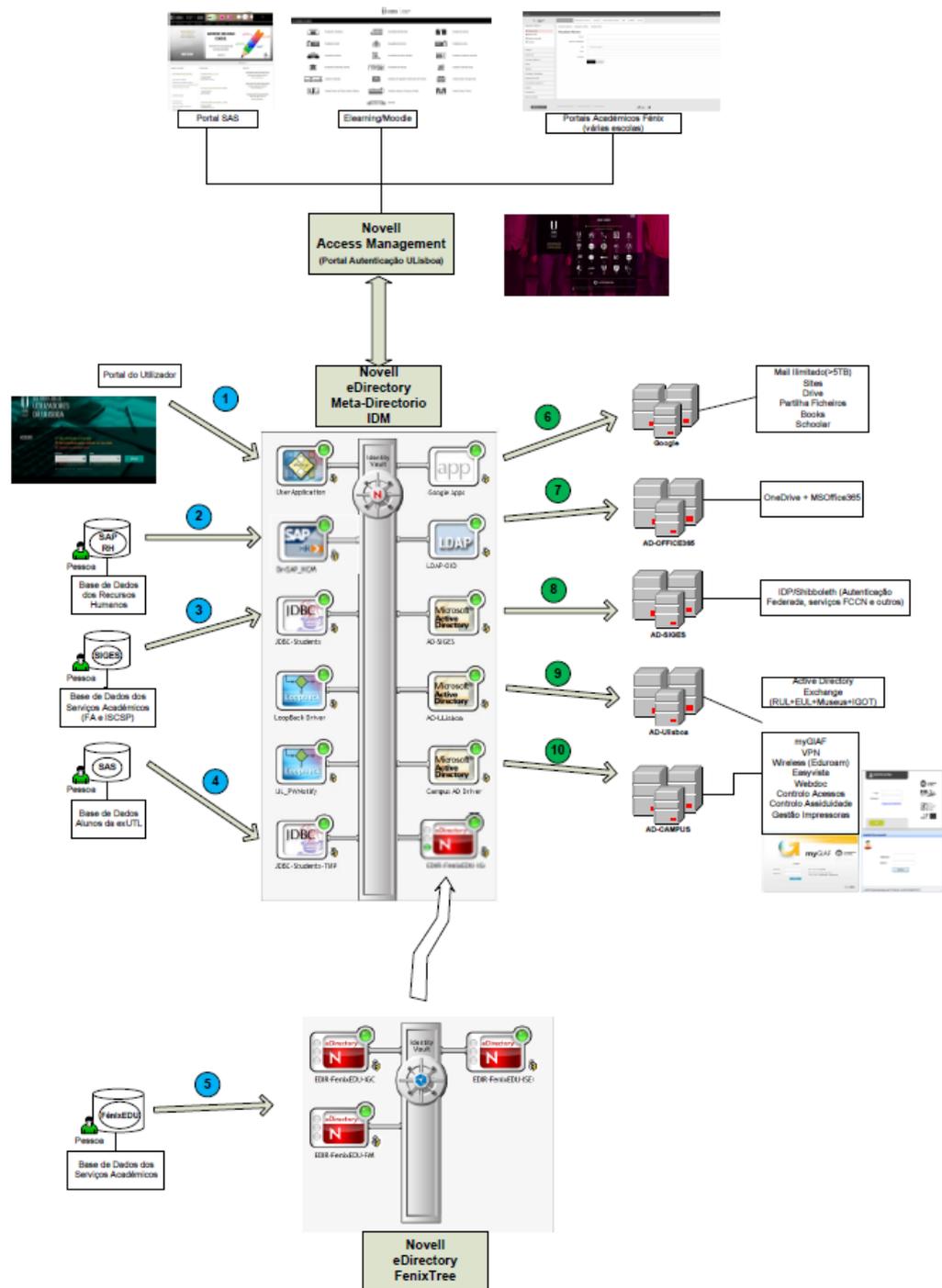


Imagem 3 – Vista geral do Sistema de Gestão de Identidade da ULisboa

A *Imagem 3* representa a organização de todo o sistema de IdM da ULisboa. Nesta figura, a coluna central representa o eDirectory Novell, onde estão guardados os

dados relativos aos vários utilizadores da universidade. À esquerda encontram-se as fontes de dados onde reside o negócio, ou seja, onde estão implementadas as regras de aprovisionamento e desaprovisionamento de utilizadores (estão representadas as bases de dados académicas e de Recursos Humanos de onde são gerados os eventos para serem processados por todos os conetores existentes no meta-diretório). Por sua vez, à direita, os serviços que dependem deste diretório, como servidores de e-mail, eduroam, VPN, MS Office365, *Google for Education*, Portal do Funcionário, Portais Académicos, entre outros. A secção superior da imagem corresponde à plataforma de Gestão de Acessos que interliga diretamente com o IdM e permite que um utilizador que possua uma ou mais contas institucionais, possa aceder aos vários serviços independentemente da conta que utilize.

Ao nível de arquitetura, o IdM é composto por três servidores, onde a camada *Identity Manager* (que contém a instalação dos vários conetores que ligam aos mais variados sistemas) está balanceada em dois dos três servidores (ver *Imagem 4*). A base de dados LDAP, que contém armazenada todos os tipos de objetos, está presente nos três servidores que se encontram constantemente sincronizados e em sistema de replicação (um dos servidores assume o papel de “*master*”, os restantes dois servidores os papéis de “*slave*”).

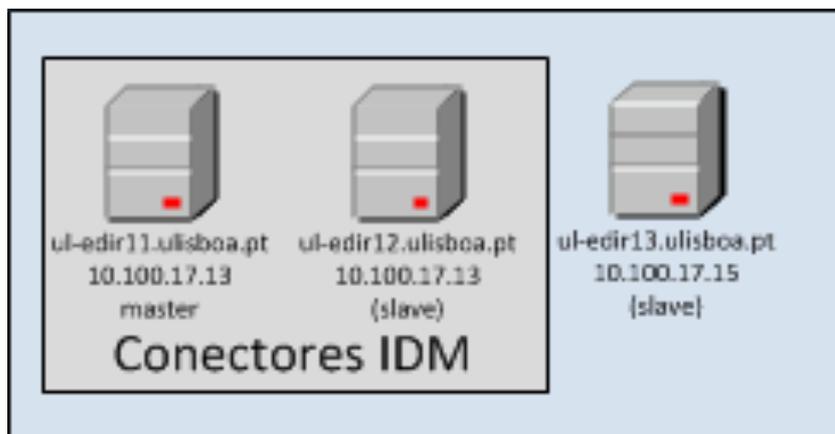


Imagem 4 - Arquitetura do eDirectory (IdM na ULisboa)

Todos os conetores implementados estão presentes nos dois primeiros servidores. Quando é feito o *deploy* de um novo conector, automaticamente o sistema replica esse novo conector para o segundo servidor. De ressaltar que apenas um conector pode estar ligado a processar eventos.

Dada a dimensão e quantidade de conectores existentes no meta-diretório, poderá ser necessário instalar a componente IDM no 3.º servidor (que até ao momento apenas contém a componente eDirectory) para um melhor balanceamento de carga ao nível de conectores ligados em cada servidor. O crescimento do número de servidores é proporcional à complexidade implementada no que respeita ao número de conectores existentes com o número de eventos despoletados pelos vários sistemas *versus* capacidades ao nível de CPU e memória dos servidores.

2.4.1 Perfilagem

Em todo o universo da Universidade de Lisboa, é muito comum que um determinado indivíduo possua um ou mais perfis em uma ou mais Unidades Orgânicas da ULisboa.

Para que seja possível fazer a gestão de identidade das várias pessoas, foi necessário definir um conjunto de classes e atributos de forma a poder perceber que determinada pessoa possui um determinado perfil em várias faculdades. Contudo, para que seja possível realizar uma sincronização coerente de toda a informação existente, é indispensável que as várias fontes de informação – Bases de Dados de Alunos ou de Recursos Humanos de cada Unidade Orgânica – contenham toda a informação atualizada sobre determinado utilizador.

O tipo de perfilagem de cada utilizador rege-se pela seguinte tipologia:

- Aluno:
 - *Licenciatura, Mestrado, Doutoramento ...*
 - *Erasmus*
 - *Regime Livre*
- Docente:
 - *Catedrático, Auxiliar, Convidado ...*
- Funcionário:
 - *Técnico/Administrativo*
 - *Bolseiro*
 - *Investigador*

Adicionalmente, como cada Unidade Orgânica da Universidade de Lisboa é independente e decide os serviços que disponibiliza aos seus utilizadores, foi necessário

implementar um *schema* que permitisse uma abordagem flexível no aprovisionamento e desaprovisionamento de serviços, conforme ditam as regras de negócio do Sistema Académico (para Alunos) e do Sistema de Recursos Humanos (SAP-HR - Pessoal Docente e Não-Docente) de cada Unidade Orgânica.

Tendo em conta a multi-perfilagem, houve a necessidade de criar classes auxiliares e um conjunto de atributos (ver anexo II e III) pertencentes a essas mesmas classes para que o Sistema de Gestão de Identidade possa identificar se determinado utilizador já existe ou não no Sistema.

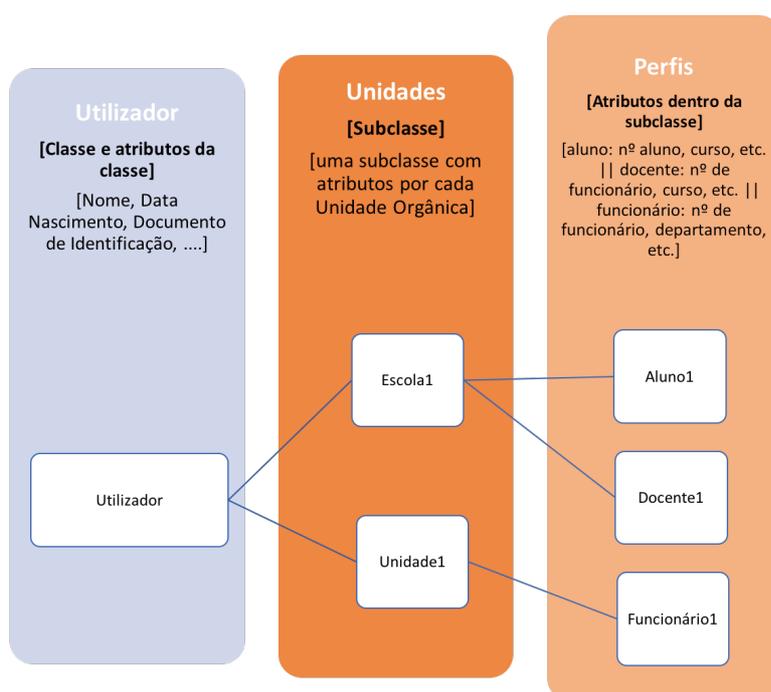


Imagem 5 - Atribuição de perfilagem de um utilizador na ULisboa

A *Imagem 5* lustra genericamente como é tratada a perfilagem de um utilizador com multi-perfis.

Existe um outro perfil menos usado, mas bastante útil no que respeita aos utilizadores “externos” ou “temporários” a uma determina Unidade Orgânica. São utilizadores que necessitam de aceder a alguns serviços (como por exemplo wi-fi eduroam ou serviço de VPN) por um determinado período de tempo. São exemplo os utilizadores de empresas que dão suporte a alguns sistemas usados por essa Instituição/Unidade, ou de utilizadores de entidades externas que têm um protocolo de cooperação com essa mesma Instituição/Unidade. Todas as contas para utilizadores

“temporários” são criadas manualmente via Portal do Utilizador. Esta funcionalidade dentro do Portal de Utilizador está disponível apenas a um grupo restrito de funcionários com privilégios de acesso especiais.

2.4.2 Schema e Atributos

Os perfis principais, dentro de cada subclasse correspondente a cada Unidade/Escola, são identificados com recurso a variáveis booleanas do diretório, e permitem identificar se um determinado perfil se encontra ativo naquela Unidade/Escola ou não. É nas várias subclasses `ULAuxFac[...]` (onde “[...]” é substituído pela sigla da unidade), que são definidos um conjunto de atributos por cada Escola/Unidade Orgânica (por exemplo `ULAuxFacFD` e `ULAuxFacRUL`, que contêm todos os atributos, valorados e não valorados, dos utilizadores na Faculdade de Direito e Reitoria respetivamente). São nestas subclasses que são definidos os atributos booleanos necessários para identificar o perfil de utilizador na Escola/Unidade (ver Anexo III).

Os atributos identificadores de perfil são:

- **Aluno (`ULisStudent[...]`)** – por exemplo `ULisStudentFL = True` (utilizador é aluno na Faculdade de Letras);
- **Docente (`ULisTeacher[...]`)** – por exemplo `ULisTeacherFC = True` (utilizador é docente na Faculdade de Ciências);
- **Funcionário (`ULisEmployee[...]`)** – por exemplo `ULisEmployeeRUL = True` (utilizador é funcionário na Reitoria);

Os atributos aplicáveis a cada perfil só estarão presentes/preenchidos no diretório caso a respetiva variável se encontre com o valor “True”. A nomenclatura é formada pelo nome do atributo concatenado com a sigla da Unidade/Instituição. As siglas que o par “nome do atributo + sigla da Unidade /Escola” pode tomar, estão representados na tabela no Anexo II.

Tal como referido no subcapítulo anterior (subcapítulo 2.4.1), é necessário também um atributo que identifique utilizadores “externos” ou “temporários”. Este tipo de perfilagem é necessária para as Escolas/Unidades em situações de criação de contas temporárias para um determinado utilizador que embora na presente data seja externo, mas que já possuiu conta na ULisboa (ou por já ter sido aluno ou desempenhado

funções em outro organismo da ULisboa). Nestes casos quando são validadas as regras de *matching* (ver secção 2.4.3), poderão ocorrer transições de registos do repositório “Temporals” para “Users”, ou de “Externos” para “Temporal”. Importa também referir que no momento de criação deste tipo de contas, é inserido uma duração/validade da conta “temporária”.

Os atributos necessários para este tipo de contas são:

- **Externo/Temporário (ULisTemporal[...]):** por exemplo
ULisTemporalFBA = True (utilizador é “temporário”/”externo” na Faculdade de Belas Artes);
- **Data de Expiração (ULExpirationDate[...]):**
ULExpirationdateFBA = 20160218 (no formato AAAAMMDD, informação introduzida pelo administrador na criação da conta Temporal);

No repositório “Temporals” os registos têm um *schema* próprio e simplificado. Ao ser criado um novo utilizador num dos sistemas que efetue *matching* com um “temporário”, o registo passa para o repositório “Users”. No entanto é necessário manter a informação de que permanece como “Temporal” naquela Escola/Unidade (a menos que a Escola/Unidade coincida).

2.4.3 Regras de *Matching*

Os vários conectores que constituem o IdM processam todo o tipo de eventos despoletados pela parametrização e programação dos vários sistemas. Sempre que um conector processa um determinado evento, são percorridas um conjunto de regras de *matching* em que o sistema valida se determinado utilizador já existe ou não no IdM. Foram implementadas duas regras principais de *matching*:

- A primeira olhando para o atributo ULBI (que possui a informação do documento de identificação);
- A segunda olhando para os atributos *Given Name* + *Surname* + *ULBirthDate*;

A necessidade da existência de uma segunda validação teve como principal motivo a existência de muitos utilizadores com nacionalidade estrangeira e com os seguintes tipos de Perfil – Alunos Erasmus, Professor Convidado, Investigadores, entre

outros. Como na maioria das vezes estes tipos de utilizadores apresentam como documento de identificação a Autorização Provisória de Residência, que é válida apenas como curta duração de tempo, ou Passaporte (validade 5 anos), sempre que há um prolongamento na Autorização de Residência, é apresentado um novo documento de identificação. Para colmatar este problema, e para evitar a criação de contas duplicadas para determinado indivíduo, foi implementado esta regra que ao validar o primeiro nome, apelido e data de nascimento diminui substancialmente o número de casos de criação de contas duplicadas.

As situações em que “primeiro nome + último nome + data de nascimento” obtêm correspondência no diretório, são tratadas individualmente e manualmente pelo colaborador administrador com privilégios para tal, após alerta do sistema. É o colaborador que após análise confirma a correspondência com algum utilizador do sistema, ou no caso de decidir que não existe correspondência, dá continuidade à criação de novo utilizador.

2.4.4 Regras de Aprovisionamento e Desaprovisionamento

Antes da explicação propriamente dita das regras de aprovisionamento e desaprovisionamento de utilizadores, é importante referir os principais diretórios existentes na árvore do eDirectory. De momento, na ULisboa os utilizadores estão divididos em quatro diretórios (*repositórios*):

- **Pending:** contas dos utilizadores ainda com o pré-registo não realizado (utilizadores que ainda não escolheram nome de utilizador, nem a senha de acesso);
- **Users:** contas dos utilizadores com pré-registo já realizado e com um ou mais perfis ativos (alunos, docentes ou funcionários).
- **External:** conta dos utilizadores sem nenhum perfil ativo;
- **Temporal:** Utilizadores unicamente com perfil “temporário” ativo;

As contas são movidas entre os vários diretórios conforme o aprovisionamento ou desaprovisionamento nos sistemas fonte (BD Académica e RH). A ação de aprovisionar e desaprovisionar determinado utilizador depende exclusivamente das regras de negócio usadas nos Sistemas Académicos e de Recursos Humanos. Não basta criar um aluno no Sistema Académico para que automaticamente seja despoletada a criação de conta, é

necessário que cumpra um conjunto de requisitos para que se considere determinado aluno ativo ou inativo. O mesmo acontece para utilizadores com perfil de Docente ou Funcionário. No sistema de RH é necessário que o utilizador esteja num código de situação ativo.

Conforme a localização da conta nos diversos repositórios, é feita a autorização para acesso/não acesso aos serviços disponibilizados pela ULisboa. Sempre que os conectores tratam eventos do tipo “move”, é despoletada uma série de ações para que os conectores sincronizem esta informação para os vários serviços (ADs, *Google Education*, MS Office365, ...) conforme as regras definidas em cada conector.

Para uma melhor perceção, na *Imagem 6* está exemplificado o ciclo de passagem dos utilizadores entre repositórios.

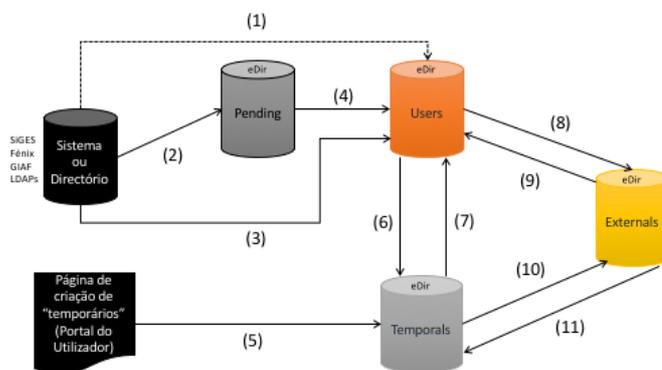


Imagem 6 – Passagem entre repositórios (numeração descrita no texto)

Para que sejam despoletadas ações nos conectores, é necessário que existam alterações em atributos específicos. Estas alterações nos atributos são feitas na grande maioria pelos sistemas nos atributos do eDirectory. A atribuição dos acessos aos sistemas por parte dos utilizadores é definida pelo repositório onde o utilizador se encontra (subsequentemente pelos atributos que tem preenchidos).

Os pontos seguintes descrevem as regras de aprovisionamento e desaprovisionamento seguidas pelo IdM da ULisboa, de acordo com a numeração da *Imagem 6*:

(1) Aprovisionamento automático

Aprovisionamento automático (sem intervenção manual dos operadores do sistema IdM). Acontece quando o registo do utilizador é criado num dos sistemas nucleares que são responsáveis pela gestão dos perfis dos Utilizadores na ULisboa (sistemas Académicos, Recursos Humanos e diretórios de Utilizadores de algumas Escolas).

(2) Aprovisionamento Campus

Um utilizador proveniente de um sistema ou diretório nuclear, responsável pela gestão dos perfis de aluno, funcionário ou docente, é aprovisionado no Sistema de Gestão de Identidades no estado pendente (Pending) caso um dos perfis passe a ativo e não haja correspondência com outro registo já existente. Este registo é criado com um nome temporário, baseado no documento de identificação, até que o utilizador aceda e registe a sua conta, escolhendo um nome definitivo.

(3) Aprovisionamento com matching

Um utilizador proveniente de um sistema ou diretório nuclear, responsável pela gestão dos perfis de aluno, funcionário ou docente, é-lhe adicionado um sub-perfil no seu registo já existente no IdM (Users) caso haja uma correspondência ao nível do número de documento de identificação. Uma subclasse com os atributos desse perfil e dessa Escola é adicionada ao registo já existente.

(4) Aprovisionamento Campus

Um registo de utilizador transita de estado pendente (Pending) para utilizador definitivo (Users) quando o utilizador conclui o seu processo de ativação de conta e escolhe um nome de utilizador e senha definitivos.

(5) Registo “temporário” sem matching

Um utilizador é criado como temporário (Temporal) no IdM quando é registado manualmente como tal através do Portal do Utilizador por um

colaborador com permissões de administração. A conta transita para o estado temporário depois de o utilizador ativar a mesma. Um registo temporário tem acessos limitados a serviços na ULisboa e possui um prazo de validade na conta que é definido de base para 3 meses.

(6) User → Temporal (sem perfis)

Caso um registo de utilizador deixe de ter perfis ativos de aluno, docente ou funcionário, o mesmo irá transitar da árvore de utilizadores ativos (Users) para temporários (Temporal) caso após inativação de todos os seus perfis se verifique que ainda possui pelo menos um perfil Temporário numa Unidade Orgânica, e que esse perfil ainda se encontre dentro do prazo de validade.

(7) Temporal → User (aprovisiona perfil)

Um utilizador proveniente de um sistema ou diretório nuclear, responsável pela gestão dos perfis de aluno, funcionário ou docente, transita da árvore de temporários (Temporal) para utilizadores ativos (Users) quando existe uma correspondência desse utilizador com um registo já existente enquanto temporário. A conta do utilizador irá neste caso passar a ter maiores privilégios de acesso aos serviços disponibilizados pela ULisboa.

(8) Desaprovisionamento User

Caso um registo de utilizador deixe de ter perfis ativos de temporário, aluno, docente ou funcionário, o mesmo irá transitar da árvore de utilizadores ativos (Users) para utilizadores externos (External) e irá perder todos os privilégios de acesso aos serviços, consoante as regras definidas em cada um, podendo não ser uma perda imediata dos acessos (exemplo: Portal de Recursos Humanos para consulta de recibos de vencimento).

(9) Reaprovisionamento User

Um utilizador proveniente de um sistema ou diretório nuclear, responsável pela gestão dos perfis de aluno, funcionário ou docente, transita da árvore de utilizadores externos (External) para utilizadores ativos (Users) quando existe uma correspondência desse utilizador com um registo já existente enquanto

externo. A conta será reativada e o utilizador passará a ter os privilégios de acesso correspondentes às regras aplicáveis ao seu Perfil e Unidade Orgânica na ULisboa.

(10) Desaprovisionamento Temporal

Caso o prazo de validade de um registo de utilizador temporário expire, a conta irá transitar da árvore de utilizadores temporários (Temporal) para utilizadores externos (External) e irá perder os privilégios de acesso limitados associados a esse tipo de contas (exemplo: acesso wireless).

(11) Reaprovisionamento Temporal

Caso a um registo de utilizador externo (External) seja adicionado um prazo de validade associado a um perfil temporário numa Unidade Orgânica, o mesmo irá transitar para a árvore de utilizadores temporários (Temporal) e passará a ter acesso limitado a serviços.

2.4.5 Conectores IdM

A solução Identity Management da NetIQ contém um conjunto de conectores pré-definidos para interligar ao mais variado tipo de Sistemas: conectores para Base de Dados (por exemplo, mySQL, JDBC Oracle, SQL Server), conectores de LDAP (por exemplo eDirectory, Active Directory, OpenLDAP, entre outros), conectores PBX, conectores para definição de políticas *standard* para centralizar um conjunto de regras básicas para todo o tipo de conectores (conectores “Loopback”), conectores para tratamento de alarmística, conectores de ligação ao *Google for Education* ou MS Office 365, entre outros. Todos estes tipos de conectores usam políticas e regras construídas via linguagem XPath (baseada em XML), que permite pormenorizar e especificar um conjunto de regras de forma a ser possível sincronizar informação entre os vários sistemas e serviços que a Universidade de Lisboa disponibiliza.

Na *Imagem 7* todos os conectores possuem dois canais:

- **Canal Publisher** – São definidas regras que permitem alterar a informação presente no IdM (designado na *Imagem 7* como *Identity Vault*,

que é o mesmo que o repositório onde reside toda a informação – eDirectory tree);

- **Canal Subscriber** – são definidas regras de sincronização e modificação de objetos, do lado do serviço/aplicação que se pretende, durante ou após, a modificação de determinada informação do utilizador ao qual o evento se refere.

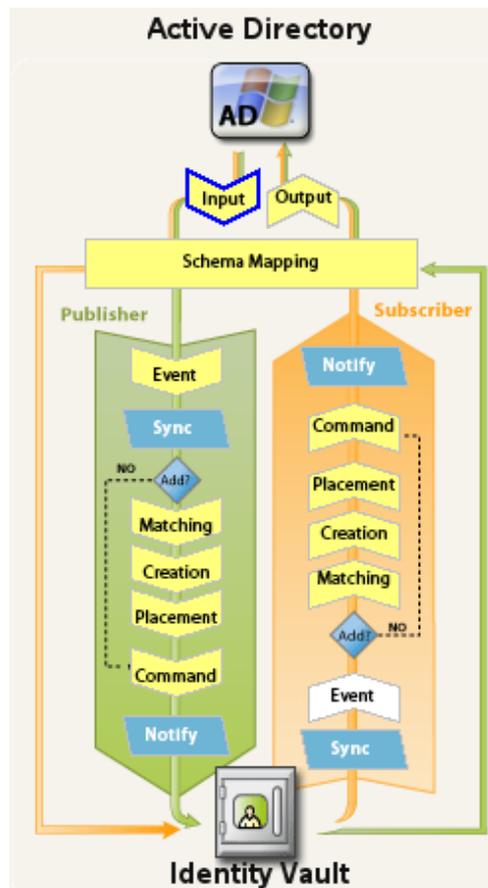


Imagem 7 – Conector do IdM que liga à AD – Active Directory

Cada tipo de conector tem um conjunto de políticas e regras base. Uma política pode conter uma ou mais regras, sendo que cada regra é constituída por um conjunto de condições e ações.

Estes tipos de conectores estão ligados aos vários sistemas e processam todo o tipo de eventos que são despoletados: inserção/alteração de dados numa BD, criação de utilizador/modificação de atributos de determinado objeto no LDAP, sincronização de senhas de acesso entre os vários LDAPs, envio de e-mails tendo por base um conjunto prévio de condições, adição/remoção de utilizadores em grupos, mudança de repositório

na estrutura de LDAP de determinado objeto segundo as regras de perfilagem, pedidos de recuperação de senha...

Sempre que um evento é lido e sincronizado com o IdM, o evento vai ser lido por todos os conectores e cada conector apenas trata o respetivo evento caso as condições existentes nas regras nele implementadas forem satisfeitas (Secção “*Event*” – visível na *Imagem 7*). Nesse caso, e antes de percorrer todas as políticas/regras de criação e modificação de informação presentes nos dois canais de cada conector, existe uma secção de “*Matching*”, que tal como o próprio nome indica, residem as regras de *matching* (mencionadas no subcapítulo 2.4.3) em que o conector vai tentar identificar a que tipo de utilizador o evento se refere (percorrendo as regras de *matching*). Caso o utilizador a que o evento se refere já exista no IdM, apenas são percorridas as políticas e regras presentes nas secções “*Command Transformation Policies*”. Caso contrário, são percorridas as políticas e regras presentes na secção “*Creation*” e “*Placement Policies*”, respetivamente, e só depois as regras da secção “*Command*”.

Seguidamente são explicadas de forma sumária algumas das políticas⁴ referidas na *Imagem 7*.

- **Event Transformation policies:** nesta secção são introduzidas as políticas que permitem transformar os eventos aplicando um filtro de diretório ou de tipo de evento (add/modify/delete). Alterar um evento inclui executar uma determinada ação com a finalidade de alterar um objeto nos diretórios de origem ou de destino, mas sem que seja adicionado nada ao evento da operação corrente.
- **Matching policies:** estas políticas são usadas para procurar um determinado objeto no repositório de destino que corresponda a um objeto não associado no repositório de origem. É de notar que estas políticas nem sempre são necessárias; no caso da ULisboa, tal como explicado no subcapítulo 2.4.3, estas políticas são essenciais para evitar a criação de contas duplicadas para cada indivíduo.

⁴ Página da NetIQ

<https://www.netiq.com/documentation/idm402/policy/data/policytypes.html>

- **Creation policies:** neste setor são adicionadas políticas que definem as condições que devem ser cumpridas para se criar um novo objeto no meta-diretorio. Existem em ambos os canais *Subscriber* e *Publisher* podendo ser diferentes. Normalmente são usadas para:
 - ✓ Vetar a criação de objetos que não cumpram com determinadas condições como por exemplo, a falta de um atributo específico;
 - ✓ Escrita de atributos *default* que permitem despoletar outros eventos para outros conectores processarem esses mesmos eventos;
 - ✓ Atribuição de senhas aleatórias no momento de criação dos objetos;
- **Placement policies:** são políticas que determinam onde os novos objetos irão ser criados e o nome que esses objetos irão ter, quer no sistema IDM, quer no sistema de destino. É necessário existir este tipo de políticas no canal *Publisher* se desejamos que a criação do objeto ocorra no IDM. Já no canal *Subscriber*, pode não ser obrigatório a existência deste tipo de políticas dependendo da natureza do sistema de destino.
- **Command Transformation policies:** estas políticas são úteis sempre que se pretende alterar os comandos que o *Identity Manager* está a enviar para o sistema de destino: por exemplo, intercetando comandos Eliminar e substituindo-os por um comando Modificar, ou substituir o comando Mover pelo comando Desativar. Nos termos mais gerais, estas políticas são usadas para alterar comandos que o *Identity Manager* executa como resultado do processamento padrão de eventos que foram submetidos ao motor que gere o meta-diretório.
- **Schema Mapping policies:** nesta secção é introduzido o mapeamento entre *schema* do meta-diretorio IDM e o *schema* do sistema de destino (por exemplo, Active Directory, campos de uma tabela de uma base de dados ou preenchimento de atributos na conta *Google for Education*);

Existe ainda a seguinte política não descritas na *Imagem 7*:

- **Output Transformation policies:** lidam principalmente com a conversão de formatos de dados que o motor do meta-diretório fornece ao servidor de destino. Exemplos dessas conversões incluem:
 - ✓ Conversão de formato do valor do atributo;
 - ✓ Conversão de vocabulário XML;
 - ✓ Colocação de mensagens de estado de retorno para serem processadas pelo motor do meta-diretório para o sistema de destino.

2.4.6 O portal do utilizador no IdM

O Portal do Utilizador é uma peça fulcral na arquitetura do Sistema de Gestão de Identidade (IdM). É neste portal que o utilizador pode consultar alguma informação pessoal, consultar todo o organograma do seu Organismo (no caso de estarem mapeados os vários colaboradores pelos vários departamentos), alterar a senha de acesso pessoal da conta (que conseqüentemente é sincronizada para todos os serviços que a ULisboa disponibiliza), etc.

O próprio produto da NetIQ possui esta componente essencial – o *User Application*. Contudo, é um produto bastante limitado no que respeita às funcionalidades, *layout* e opções disponíveis no menu de navegação apresentadas ao utilizador. Na análise feita ao componente disponibilizado pela solução, compreendeu-se que a mesma não satisfazia a realidade existente na ULisboa, devido a complexidade existente na diversidade de perfisagens, nas de regras de aprovisionamento e desaprovisionamento de utilizadores, na atribuição de serviços conforme a Unidade Orgânica e na flexibilidade na adição e remoção de funcionalidades com vista a uma mudança de estratégia num futuro próximo em paralelismo com os serviços que cada Instituição pretende ter disponível para os seus utilizadores.

2.5 Modelo Sequencial Linear

É consoante as especificidades do projeto que o modelo de procedimento de engenharia de *software* é escolhido. Essa escolha assenta na natureza do projeto, nos métodos que vão ser aplicados, nas ferramentas a serem utilizadas e no calendário de entregas que é exigido pelo cliente.

Para a implementação do projeto do PU recorreu-se a um modelo de desenvolvimento de *software* sequencial linear, também chamado de ciclo de vida

clássico, ou abordagem “top-down”. Este modelo foi proposto por Royce em 1970 (Royce, 1970).

Neste modelo de desenvolvimento de *software* é sugerida uma abordagem sistemática e sequencial para o desenvolvimento de *software*, que se inicia ao nível da análise, desenho, código, testes e suporte, como apresentado na *Imagem 8* (Pressman & Maxim, 2014).

Neste modelo, a passagem de uma fase para outra apenas é feita apenas quando a antecedente termina.

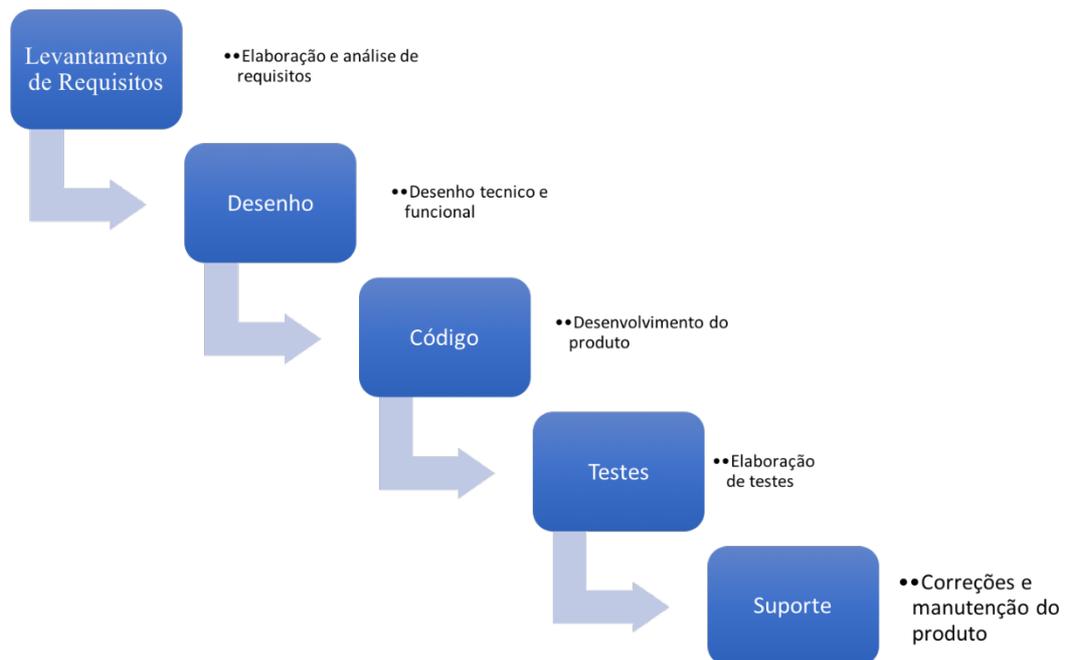


Imagem 8 - Modelo sequencial linear Adaptado de (Pressman & Maxim, 2014)

As diferentes etapas do modelo *waterfall*, descrito na *Imagem 8* são:

- **Análise e definição dos requisitos**

Nesta etapa o levantamento de requisitos é focado no *software*, ou seja, estabelecem-se os requisitos do produto que se deseja desenvolver, a finalidade, o comportamento, o desempenho e as funcionalidades necessárias das interfaces.

- **Desenho técnico e funcional**

O *design* de *software* é, na verdade, um processo de múltiplos passos que se concentra em quatro atributos distintos de um programa: estrutura de dados, arquitetura de *software*, representações de interface e detalhes processuais (algorítmicos). O

processo de desenho traduz os requisitos numa representação do *software* que pode ser avaliado quanto à qualidade antes do início da programação. Como os requisitos, o desenho é documentado e faz parte da configuração do *software*.

- **Implementação**

É nesta fase que todo o trabalho técnico e funcional é traduzido para código, ou seja, implementado. Sugere-se neste modelo que no início seja incluído um teste unitário nos módulos desta etapa para as unidades de código produzidas serem testadas.

- **Testes**

Após a codificação inicia-se o programa de testes, que se centra em dois pontos essenciais: os componentes internos do *software* (testes aplicativos) e as funcionalidades externas (testes funcionais). Desta forma é assegurado que todas as funções foram testadas, e que produzem os outputs esperados, coincidentes com os requisitos especificados.

- **Suporte**

É nesta etapa que é feita a correção de erros que não foram detetados na implementação. O *software* após a entrega tem de estar adaptado para acomodar mudanças no seu ambiente externo (por exemplo mudança de dispositivo periférico, melhorias funcionais necessárias para o cliente, aumento de desempenho). O suporte/manutenção de *software* reaplica cada uma das fases anteriores para um programa existente em vez de um novo.

O modelo sequencial linear é o paradigma mais antigo e o mais utilizado para engenharia de *software*. No entanto, já foram levantadas várias questões relativamente à sua eficácia.

Algumas das questões que são levantadas advêm do facto de que em projetos reais o fluxo sequencial nem sempre é cumprido, apesar de o modelo linear poder acomodar essa iteração, é feito de forma indireta, podendo causar confusão à medida que a equipa do projeto prossegue.

Uma outra variável indesejada que é apontada, é a incapacidade apontada à maioria dos clientes para indicar todos os requisitos explicitamente. O modelo

sequencial linear requer a premissa de um levantamento completo de requisitos iniciais, e nem sempre o projeto consegue acomodar a incerteza natural que existe no início de muitos projetos.

Outra lacuna que é apontada ao projeto é a impaciência por parte dos clientes, pois uma versão inicial do trabalho / programa, apenas estará disponível no final da interação (consequentemente projeto). Um grande erro, se não for detetado até o programa de trabalho ser revisado, pode ser desastroso.

Neste tipo de abordagem de *software* também podem ocorrer bloqueios por parte de alguns membros da equipa do projeto, por estarem dependentes da finalização das tarefas por outros elementos

O ciclo de vida clássico continua a ser um modelo processual amplamente utilizado para engenharia de *software*. Embora tenha deficiências, é significativamente melhor do que uma abordagem aleatória para o desenvolvimento de *software*.

Capítulo 3

Tecnologias Utilizadas no Portal do Utilizador

A escolha de um *framework* de Desenvolvimento, num mundo cheio de opções, nem sempre é fácil a decisão de escolher uma.

Um *framework*, em desenvolvimento de *software*, é um conjunto de bibliotecas bem estruturadas que realizam uma função bem definida (Framework, s.d.). Um *framework* destina-se a aliviar a sobrecarga associada a atividades comuns realizadas em desenvolvimento de *software*, como por exemplo, o acesso a bases de dados, modelação de dados, controle da sessão, permitindo uma reutilização de código.

Um *framework* captura a funcionalidade comum a várias aplicações, pois na maior parte das aplicações existem funcionalidades que são partilhadas - pertencem a um mesmo domínio de problema.

A escolha de um *framework* é uma decisão agonizante (Allen, September 2008). A principal questão não está só em escolher um *framework*, já que muitas partilham a mesma funcionalidade requerida: o programador, quando confrontado com a necessidade de implementar uma funcionalidade, deve ter em conta os *frameworks* que o podem ajudar, tendo em conta que uns são amplamente utilizados, outros promissores, outros são as tendências de mercado do momento, e até quais as tecnologias com que o programador se identifica. Em particular, um maior número de funcionalidades não implica que o *framework* é o mais indicado, pois essa complexidade pode atrapalhar o programador e ser prejudicial à criatividade.

Para a escolha do *framework* utilizada na implementação do novo Portal do Utilizador, foi elaborada uma pesquisa dos *framework* de desenvolvimento de *software* web mais conhecidos disponíveis no mercado. Para esta avaliação, foram tidos em conta os custos de aprendizagem para a equipa de desenvolvimento que desenvolveu o Portal de Utilizador, a documentação disponibilizada, a comunidade, a continuidade e o

suporte do projeto no futuro. Após esta análise, foi decidido pela equipa de projeto de implementação do sistema de Gestão de Identidade na Universidade de Lisboa, a adoção do *framework* normalmente utilizada no desenvolvimento de *software* pelo Núcleo de Desenvolvimento de Software – NDS/DI: o JBoss Seam. Este *framework* não apresenta custo de aprendizagem, tem as funcionalidades/tecnologias necessárias para a implementação e agilização do desenvolvimento e portanto apresentar um menor risco de a implementação do Portal de Utilizador falhar.

O JBoss Seam é uma poderosa plataforma de desenvolvimento *open source* para desenvolvimento de aplicações web em Java EE. O Seam integra um vasto *stack* tecnologias como por exemplo o *Asynchronous JavaScript*, XML (AJAX), *JavaServer Faces* (JSF), *Java Persistence API* (JPA), *Enterprise Java Beans* (EJB 3.0), *Business Process Management* (BPM), entre outras.

O Seam foi projetado desde o início para eliminar a complexidade em níveis de arquitetura e API. Confere aos programadores total controlo sobre a implementação da lógica de negócio sem se preocupar com a exposição das informações e/ou configuração excessiva de arquivos XML, dispondo de anotações para classes Java e componentes bem definidos para a camada de apresentação.

Gavin King, líder do projeto Seam e líder da especificação da JSR 299, definiu o projeto Seam como uma API de criação de aplicações WEB de fácil manutenção e ótima produtividade. A *Imagem 9* ilustra a forma como este *framework* se integra com outros *frameworks* de desenvolvimento web em Java EE.

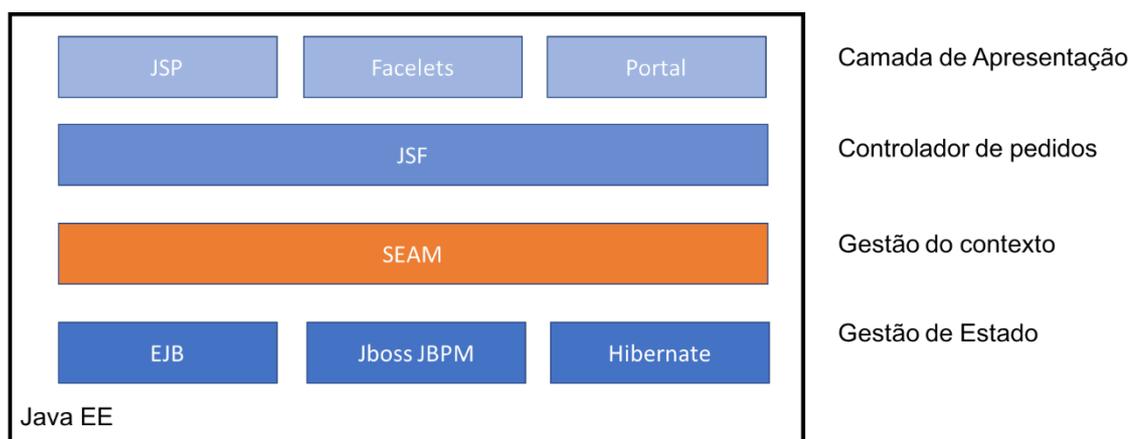


Imagem 9 - Integração do *framework* JBoss Seam numa arquitetura Java EE adaptado de ⁵

⁵ Página do JBoss Seam Framework <http://seamframework.org>

3.1 Arquitetura

A arquitetura do Portal do Utilizador divide-se em três camadas: apresentação, negócio e dados, como ilustrado na *Imagem 10*.

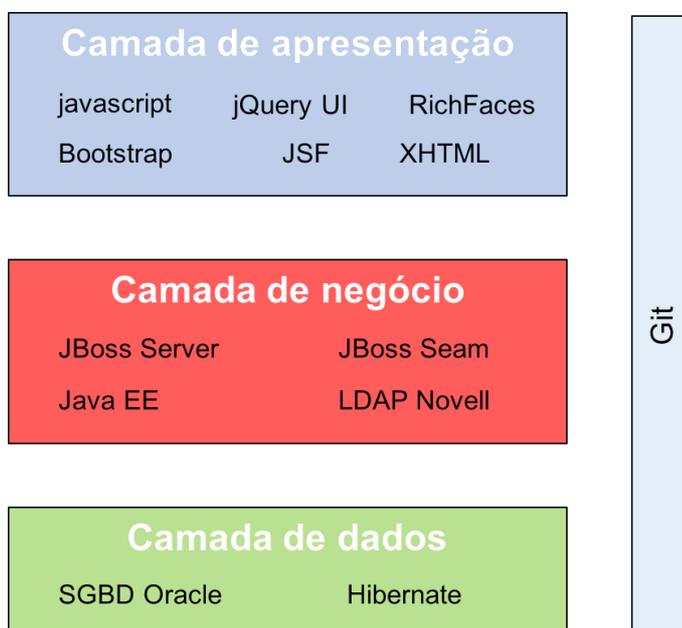


Imagem 10 – Arquitetura do Portal do Utilizador e seus componentes

Para a camada de Apresentação (*User Interface* – UI) serão utilizadas páginas JSF que exibem a interface gráfica, capturando os dados enviados via formulário e mostrando os resultados. Na interface gráfica, os objetos de apresentação são mapeados para objetos de negócio. Esta camada é responsável por:

- gerir os pedidos do utilizador e apresentar-lhe as respetivas respostas
- providenciar a delegação de chamadas à lógica de negócio
- disponibilizar os dados de uma forma coerente ao utilizador
- executar validação de dados inseridos pelo utilizador

A camada de Negócio é implementada utilizando EJB 3 *session beans* e classes java com anotações. Esta camada serve de controlador dos eventos disparados pelas páginas JSF, e é responsável por:

- implementar a lógica de negócio da aplicação
- validar dados de negócio
- implementar a gestão transacional

- permitir que as interfaces comuniquem com as outras camadas
- gerir as dependências entre objetos de negócio
- adicionar flexibilidade entre a camada de apresentação e a camada de dados de forma a elas não comunicarem diretamente
- estabelecer as ligações ao LDAP

A camada de Dados é onde vivem os objetos de negócio a serem persistidos e/ou utilizados nas regras de negócio. É esta camada a responsável pela correta gestão de todos os dados da aplicação. Mais concretamente:

- Transforma informação relacional em objetos Java. Esta funcionalidade é executada pelo Hibernate (ver *Imagem 11* e secção 3.2.8) através de uma linguagem OO chamada HQL
- Grava, atualiza e remove a informação numa base de dados

3.2 Ferramentas

Em seguida são descritas as várias ferramentas utilizadas no desenvolvimento do Portal do Utilizador, de acordo com a *Imagem 10*.

3.2.1 XHTML

Tratando-se de um projeto web, a linguagem de marcação ou *Markup*, é o XHTML. Uma linguagem de marcação (Linguagens de Marcação, s.d.) é um conjunto de códigos aplicados a um texto ou a dados, com o fim de adicionar informações particulares sobre esse texto ou dado, ou sobre textos específicos.

O XHTML é HTML escrito com *tags* XML. A especificação da linguagem XHTML descreve que os documentos web XHTML devem ser formatados como se de um documento XML se tratasse. No XHTML todas as *tags* têm obrigatoriamente de ser fechadas e escritas em *lower case*, ao contrário da linguagem HTML em que nem sempre a obrigatoriedade de fechar *tags* específicas – é caso a *tag*
.

3.2.2 JavaScript

O JavaScript (Flanagan & Ferguson, 2002) é a linguagem de programação da web. A esmagadora maioria dos *sites* modernos usam JavaScript e todos os navegadores

de internet modernos – desktops, consolas, dispositivos móveis e smartphones – incluem JavaScript, tornando o JavaScript a linguagem de programação mais omnipresente da história. O JavaScript faz parte da tríade de tecnologias que está presente na web e que todos os programadores de desenvolvimento web têm obrigatoriamente de saber: HTML, CSS e JavaScript.

O JavaScript é utilizado no projeto de implementação do PU, através de bibliotecas como o jQuery, para deteção de eventos como onChange, onMouseOver e onClick, criando efeitos visuais através da alteração das propriedades dos objetos HTML, ou mesmo executando pedidos AJAX, que permitem à página web fazer pedidos assíncronos ao servidor e atualizar o seu conteúdo dinamicamente.

jQuery UI

O jQuery⁶ é uma biblioteca JavaScript eficiente, pequena e rica em recursos que foi desenvolvida para simplificar scripts do lado do cliente que interagem com o HTML. A sintaxe do jQuery foi desenvolvida para tornar mais simples a navegação do documento HTML, a seleção de elementos DOM, criar animações, manipular eventos e desenvolver aplicações AJAX.

Google reCAPTCHA

CAPTCHA é um acrónimo da expressão “Completely Automated Public Turing test to tell Computers and Humans Apart” (Carnegie Mellon University, 2010) que se traduz por *teste de Turing público completamente automatizado para diferenciação entre computadores e humanos*. Esta ferramenta estabelece medidas de segurança que impede programas automatizados de executar tarefas em ambiente web, que apenas devem ser executadas por humanos, de forma não automática. Exemplos de situações incluem a criação de contas de e-mail, a submissão de grandes ficheiros num servidor, etc. Essas tarefas consistem em pedir ao utilizador da página web que executem uma tarefa considerada difícil ou impossível de ser executada automaticamente, como por exemplo, decifrar frases com caracteres distorcidos ou identificar determinados objetos nas figuras que são apresentadas (Ahn, Maurer, McMillen, Abraham, & Blum, 2008). Através de validações de procedimentos exigidos pela ferramenta CAPTCHA, reduz-se

⁶ Página do jQuery <http://jquery.com/>

a criação de *spam*, criação de ciclos infinitos para detecção de senhas de acesso, nomes de utilizador, acessos a contas de e-mail, entre outras.

O projeto Google reCAPTCHA⁷, que partiu de uma interface do projeto CAPTCHA, é um serviço Google que adiciona mecanismos de proteção nos sites de spam e abusos. Para isso, o reCAPTCHA recorre a técnicas avançadas de análise de risco para decifrar se é um humano ou uma máquina que está a realizar determinada ação crítica. A API disponibilizada permite facilmente que se adicione um CAPTCHA a um projeto, sem necessidade de recorrer a desenvolvimentos adicionais.

3.2.3 RichFaces

O RichFaces (RichFaces 3.3.X , s.d.) é um conjunto de componentes disponíveis numa biblioteca, que adicionam capacidades AJAX em aplicações web que utilizam o *framework* JSP, sem recorrer a JavaScript.

Os componentes do RichFaces podem ser considerados uma extensão do Ajax4jsf com inúmeros componentes com AJAX embutido e com suporte a Skins que permitem uma padronização das interfaces das aplicações.

Os componentes RichFaces são divididos em duas bibliotecas de *tags*: a RichFaces, que fornece temas (skin) e Ajax4jsf *Component Development Kit* (CDK).

3.2.4 Bootstrap

Hoje em dia, existe uma forte tendência de se recorrer aos *framework* CSS para o desenvolvimento WEB, pois permitem a utilização imediata de estilos base nas páginas. Ter classes CSS base para os elementos HTML fundamentais e *grid-view* para um projeto iniciado do zero de desenvolvimento de interfaces WEB, gera métodos que permitem melhorar, agilizar e fortalecer o desenvolvimento das interfaces.

O Bootstrap (Bootstrap 3 Tutorial, s.d.) é um *framework* de desenvolvimento de *front-end* muito popular para desenvolvimento Web, que foi criado pelo Twitter a partir de código usado internamente. Após ter sido tornado um projeto *opensource*, cresceu e ganhou muitos adeptos e importância de mercado.

O Bootstrap (Bootstrap - front-end framework, s.d.) inclui uma série de funcionalidades como uma base CSS que inclui um estilo visual coerente para a maioria

⁷ Página do Google reCAPTCHA <https://www.google.com/recaptcha>

das *tags* HTML, ícones, componentes CSS e *grids-view* prontas a serem utilizadas. Para além de todo o potencial de reaproveitamento de estilos, o *framework* possui a capacidade de tornar uma página responsiva e adaptativa consoante o dispositivo utilizado para visualização, sem a necessidade de customizar ou otimizar para cada situação. O Bootstrap⁸ também inclui alguns componentes JavaScript na forma de plugins jQuery, que fornecem elementos adicionais de interface.

3.2.5 JSF

O JavaServer Faces é uma especificação Java que define (JSR 127: JavaServer Faces, 2016) uma arquitetura e uma API, que simplificam a criação e manutenção de interfaces gráficas de utilizador para aplicações web.

Os JSF utilizam *facelets*, que podem ser definidos como secções reutilizáveis de conteúdos. Um *facelet* (What is a "Facelet"?, s.d.) é um elemento de uma visão composta que pode ser facilmente combinado com outros *facelet*.

Uma das grandes vantagens da utilização de JSF é a criação de uma abstração para o utilizador do JavaScript e do HTML, com a utilização de um modelo de interface do utilizador baseado em componentes visuais pré-prontos (arquivos XML que são *facelet*). Estes arquivos XML geram na execução da página JSF código HTML e javascript que é interpretado pelo *browser*. Desta forma, o código desenvolvido pelo programador torna-se mais limpo e organizado, com componentes ricos em funcionalidades, que agilizam o desenvolvimento e a criação de aplicações web.

3.2.6 Java EE

Java é uma linguagem de programação orientada a objetos, que tem como principal vantagem ser independente da plataforma de execução. Esta independência permite que a aplicação seja executada nos mais populares sistemas operativos.

O Java EE (O que é Java EE?, 2016) é uma plataforma que contém um conjunto de tecnologias coordenadas que reduz significativamente o custo e a complexidade do desenvolvimento, implementação e administração de aplicações de várias camadas centradas no servidor. O Java EE tem como base a plataforma Java SE e oferece um

⁸ Página do Bootstrap <http://getbootstrap.com/>

conjunto de APIs para desenvolvimento e execução de aplicações portáteis, robustas, escaláveis, confiáveis e seguras no lado do servidor.

3.2.7 LDAP Novell

A LDAP Novell é uma API Java (LDAP Classes for Java, s.d.) disponibilizada pela empresa Novell que permite desenvolver aplicações que acedem, administram e atualizam informações armazenadas no Novell eDirectory, ou em outros diretórios compatíveis com LDAP.

3.2.8 Hibernate

O Hibernate⁹ ORM (Object-relational mapping) permite aos programadores desenvolverem com maior facilidade aplicações cujos dados têm de sobreviver ao processo de aplicação. Como um sistema de mapeamento entre objetos e estrutura relacionais, o Hibernate está preocupado com a persistência de dados, uma vez que se aplica a bancos de dados relacionais (via JDBC). Na *Imagem 11* está representada a arquitetura do hibernate.

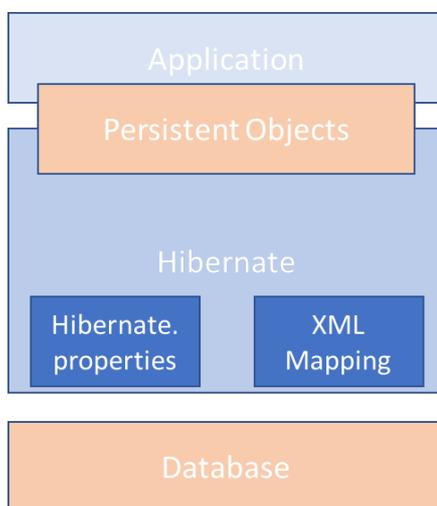


Imagem 11- Visão Global da Arquitetura do Hibernate adaptado de ¹⁰

⁹ Página Hibernate <http://hibernate.org/orm/>

¹⁰ Página da JBoss

http://docs.jboss.org/ejb3/app-server/Hibernate3/reference/en/html_single/

Este sistema (Definição de Hibernate, s.d.), além de mapear as tabelas de uma base de dados em objetos Java, também disponibiliza funcionalidades que permitem efetuar interrogações e tratar os resultados de uma forma transparente em relação à base de dados utilizada. O Hibernate gera as chamadas SQL e trata do processamento dos resultados, permitindo desde modo que a aplicação seja portátil para diversos sistemas de gestão de bases de dados SQL.

3.2.9 Sistema de gestão de bases de dados Oracle

O SGBD da Oracle¹¹ é um sistema de gestão base de dados multiplataforma de elevado desempenho, fiabilidade, confidencialidade, integridade. As soluções de base de dados Oracle disponibilizam ferramentas de *backup* e restauro dos dados, e mecanismos de acessos competitivos. A empresa Oracle é líder mundial de base de dados.

3.2.10 JBoss Seam

O JBoss Seam¹² é um *framework* para aplicações web desenvolvido pela *JBoss Application Server*, que é uma divisão da *Red Hat*, e utiliza a maior parte dos conceitos da especificação Java EE5, que têm como objetivo facilitar a integração e o desenvolvimento de aplicações empresariais. O Seam (Seam: Repenser l'architecture des applications web? , s.d.) fornece um modelo de componentes, API's e anotações para facilitar a integração de padrões Java EE 5 com *JavaServer Faces (JSF)*, *Java Persistence API (JPA)*, *Enterprise JavaBeans (EJB 3.0)*, *AJAX* e a administração de processos de negócio. A *Imagem 12* ilustra a arquitetura deste *framework*, onde se pode ver que o JBoss Seam consiste numa camada de apresentação definida em JSF, numa camada middleware (definida em EJB3, que contém os processos da lógica de negócio e os processos de processamento dos dados), e uma camada de dados que contém a informação a disponibilizar ao utilizador.

¹¹ Página Oficial da Oracle <https://www.oracle.com>

¹² Página do Seam Framework <http://seamframework.org>

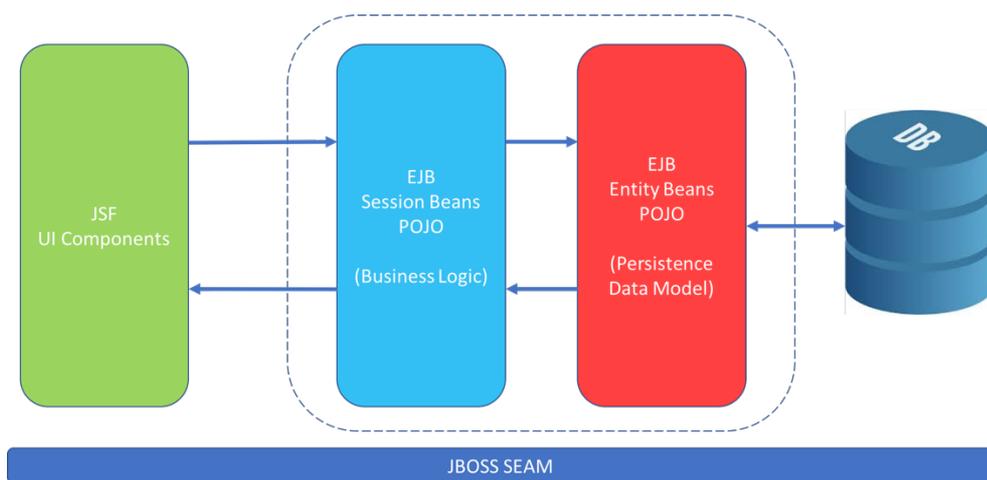


Imagem 12 - Arquitetura JBoss Seam adaptado de (*Seam: Repenser l'architecture des applications web?*, s.d.)

A camada de apresentação é composta por um conjunto de componentes JSF. Os EJB desempenham um duplo papel, pois permitem a troca de dados entre a camada de apresentação e a camada de negócios, validação de dados, gestão de transações e sessões de utilizador. As entidades EJB representam o modelo de dados, que são diretamente persistidos na base de dados através das API's de persistência.

Na *Imagem 13* estão representadas sobre esquema, um conjunto de tecnologias do stack de ferramentas do JBoss Seam, distribuídas pelas diversas camadas.

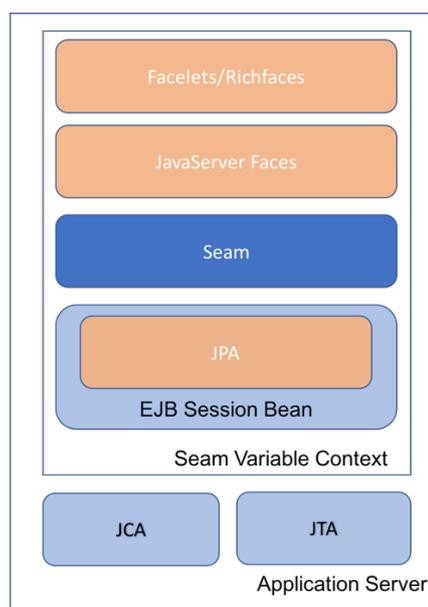


Imagem 13 - Algumas das tecnologias integradas no stack de ferramentas do Seam (*Seam: Repenser l'architecture des applications web?*, s.d.)

3.2.11 JBoss Server

Um servidor aplicacional é um tipo de servidor projetado para instalar, operar e hospedar aplicações para utilizadores finais acederem a serviços de TI e organizações (Techopedia, s.d.). Funciona também como interface entre a aplicação e o servidor de base de dados, aplicando todas as regras definidas para o acesso de cada um dos atores no nosso sistema, bem como as regras de integridade de base de dados. O acesso aos dados (quer do IdM quer da Base de dados), será feita através do servidor aplicacional.

O JBoss Server é um servidor aplicacional web que suporta o *deploy* de projetos com as tecnologias JBoss Seam e outras.

3.2.12 TestNG

O TestNG¹³ é um *framework* Java, inspirado em outros *framework* de teste como a JUnit e a NUnit. Tem integração direta com inúmeros *framework*, entre os quais os utilizados no projeto.

O objetivo principal do TestNG é abranger uma grande quantidade de abordagens de categorias de teste: unitárias, funcionais, testes a unidades, integração e testes a servidores aplicacionais externos (por exemplo, a elaboração de testes de carga). Permite também elaborar um conjunto de relatórios fáceis de interpretar com os resultados dos testes.

Para a criação de testes com o *framework*, recorre-se a classes Java preparadas com anotações TestNG, sendo o critério de execução dos métodos, ou da bateria de testes, dirigido por um conjunto de anotações Java: `@BeforeSuite`, `@AfterSuite`, `@BeforeTest`, `@AfterTest`, `@BeforeGroup` e `@AfterGroup`, `@BeforeMethod` e `@AfterMethod`. Na execução da classe TestNG, é garantido pela API do *framework* que os métodos "`@Before[...]`" são executados consoantes a ordem de herança (superclasse mais alta primeiro, depois descendo a cadeia de herança) e os métodos "`@After[...]`" na ordem inversa (subindo a cadeia de herança).

3.2.13 GIT

Uma vez que a totalidade do projeto subjacente ao IdM da ULisboa envolve várias pessoas, e que a qualquer momento pode vir a ser necessário reverter alguma alteração

¹³ Página oficial TestNG <http://testng.org/>

feita no código, todo o desenvolvimento é feito com base em controlo de versões. A ferramenta usada para o efeito neste projeto foi o Git, uma ferramenta desenhada tendo em vista a velocidade, integridade da informação e suporte para trabalho desenvolvido em sistemas (Torvalds, 2017).

As várias versões de todo código relacionado com este projeto encontram-se assim disponíveis internamente (num servidor da Reitoria da ULisboa e não num servidor público, por motivos de confidencialidade do código) para os vários programadores encarregues do desenvolvimento do IdM.

Capítulo 4

O trabalho

Antes de se iniciar o levantamento de requisitos do projeto, foi feita uma pesquisa do trabalho relacionado que existia sobre implementações de Sistemas de Gestão de Identidade. O trabalho de pesquisa ajudou-me a compreender a finalidade das implementações deste tipo de projetos e quais os objetivos que teriam de ser levados em conta para a implementação do componente do projeto do IdM, o “Portal do Utilizador”. Também permitiu a análise de casos de sucesso e insucesso de implementações de soluções idênticas.

As minhas principais tarefas nas primeiras semanas consistiram em pesquisas sobre o tema do projeto e analisar as várias soluções existentes no mercado. No entanto, dada a grande diversidade de abordagens e conceitos sobre gestão de identidade, também foi feita uma aposta na leitura de conceitos como de identidade digital e de casos de estudo de implementações de Sistemas de Gestão do Utilizador. Através dessas leituras da documentação facultada pela equipa de Projeto do IdM colmatadas com pesquisas mais livres utilizando motores de busca digitais, que consegui construir e aflorar o conhecimento sobre conceitos de gestão de Identidade, e elaborar um levantamento de riscos a ter em conta na implementação do projeto em causa.

Outra das preocupações que tida em conta, foi o enquadramento que me foi feito por parte das diversas equipas do Departamento de Informática dos Serviços Centrais sobre a infraestrutura da ULisboa, dos principais Sistemas de Informação e dos projetos que decorriam em paralelo. Esta amostragem tecnológica criou em mim uma capacidade crítica adicional para as necessidades, gerando um enquadramento tecnológico e estratégico da importância da implementação do Portal do Utilizador. Foi também caso de apreciação a antiga página do utilizador que permitia já ao utilizador um pequeno

conjunto de funcionalidades de gestão que muito facilitavam o cotidiano dos utilizadores: estudantes, não docentes e docentes.

O modelo de implementação do projeto, representado na *Imagem 8* e descrito no subcapítulo 2.5 foi o modelo utilizado para a implementação do projeto.

4.1 Análise de Utilizadores e Tarefas

Não é uma tarefa simples definir o que é a Estatística (Graça Martins, 2005). Há quem defina estatística como sendo um conjunto de técnicas de tratamento de dados, mas é muito mais do que isso! A Estatística é uma "arte" e uma ciência que permite tirar conclusões e de uma maneira geral fazer inferências a partir de conjuntos de dados. A Estatística tem vindo a amadurecer ao longo dos anos, e a necessidade de uma maior formalização nos meios utilizados, fez com que se desenvolvessem métodos e técnicas de Inferência Estatística. Assim, por volta de 1960 os textos de Estatística debruçavam-se especialmente sobre métodos de estimação e de testes de hipóteses, assumindo determinadas famílias de modelos, descurando os aspetos práticos da análise dos dados. Nas duas últimas décadas com a disponibilização e acesso a ferramentas computacionais e de processamento de dados cada vez mais poderosas, a estatística tem seguido abordagens de desenvolvimento de métodos de análise e exploração de dados.

A utilização da Estatística (Reis , Melo , Andrade, & Calapez) é abrangente, e cobre áreas como as ciências sociais, políticas, económicas, biológicas, físicas, médicas, de engenharia, entre outras. Com a transversalidade de utilização da estatística em inúmeras ciências, os métodos de amostragem e de inferência estatística tornaram-se um dos principais instrumentos do método científico.

O primeiro passo do desenvolvimento centrado no utilizador procura descobrir quem são os utilizadores do sistema, e identificar as suas necessidades ou problemas reais. (Fonseca, Campos, & Gonçalves, 2012)

A análise de utilizadores e de tarefas envolve três aspetos: os utilizadores, as suas tarefas e o ambiente, ou ambientes, em que as realizam. Para o desenvolvimento de interfaces fáceis de aprender e de usar, necessitámos de perceber as características que os utilizadores trazem para as suas tarefas, como são conceptualizadas essas tarefas antes de iniciar a interação com a nossa interface, e finalmente como é que o ambiente onde eles usam a interface para poder afetar a sua capacidade de realizar a tarefa com sucesso.

De acordo com isto, o objetivo do levantamento de requisitos é colecionar um conjunto suficiente e fidedigno de dados, de modo a produzirmos uma lista estável de requisitos, quer funcionais quer do perfil dos nossos potenciais utilizadores. (Fonseca, Campos, & Gonçalves, 2012)

Para efetuar a análise das tarefas e dos utilizadores que vão utilizar o Portal do Utilizador, foi efetuado um questionário a uma amostra significativa de utilizadores do universo da ULisboa. O questionário encontra-se reproduzido na íntegra no Anexo IV, tendo sido disponibilizado aos utilizadores sob a forma de um [formulário Google](#). Foram recolhidas 123 respostas, entre os dias 14 de junho de 2017 e 29 de junho de 2017. O tratamento dos resultados desse inquérito, recolheu informações do tipo de utilizadores, dos seus hábitos, da cultura informática e da sua capacidade de interagir com Sistemas de Informação. Com essa informação, foram gerados gráficos de interpretação de dados obtidos com cruzamento de variáveis, e feita uma leitura de cada variável analisada. No final, foi feita uma contextualização geral dos dados, e das conclusões obtidas.

4.1.1 Análise Estatística dos resultados

Os resultados obtidos com o inquérito de utilizadores foram sujeitos a uma análise estatística para compreender quem são os futuros utilizadores do PU e, desta forma, adaptar o próprio PU ao universo dos seus dos mesmos. As questões colocadas aos utilizadores são classificadas em 3 categorias diferentes:

- Questões relativas à caracterização pessoal do utilizador:
 - Sexo
 - Faixa etária
 - Tipos de utilizador (aluno, ex-aluno, funcionário docente, funcionário não-docente)
 - Instituição da ULisboa a que pertence
 - Métodos de aprendizagem preferidos
- Questões relativas ao conhecimento do utilizador sobre tecnologias de informação:
 - Tempo médio de utilização do computador
 - Serviços de e-mail usados

- Ferramentas de TI usadas
- Frequência de alteração de senhas de acesso pessoais
- Questões relativas ao Portal do Utilizador (PU):
 - Conhecimento acerca da conta Campus e seus serviços
 - Utilização da conta Campus
 - Frequência de alteração da senha de acesso à conta Campus
 - Local e dispositivo onde é usada a conta Campus
 - Funcionalidades desejadas no PU

Em seguida apresentam-se os resultados associados a cada uma destas perguntas, incluindo cruzamento entre questões. Na secção 4.1.2 far-se-á uma síntese geral dos resultados, incluindo algumas considerações finais que podem ser retiradas a partir dos mesmos e na relação com a caracterização dos futuros utilizadores do PU.

Sexo e Faixa etária

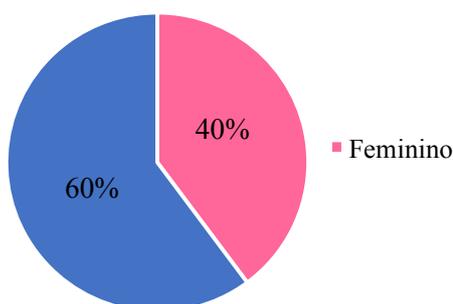


Imagem 14 – Sexo dos participantes

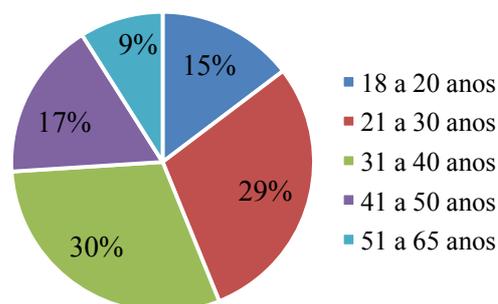


Imagem. 15 – Faixa etária dos participantes

As imagens *Imagem 14* e *Imagem. 15* mostram, respetivamente, a distribuição por sexo e por faixa etária dos 123 participantes no inquérito. Como se pode observar, a maioria é do sexo masculino (60%). As idades dos participantes situam-se tendencialmente na faixa dos 21 aos 30 anos e na faixa dos 31 aos 40 anos (respetivamente, 29% e 30%). A faixa etária com menor percentagem de participantes é a dos mais velhos (entre os 51 e os 65 anos de idade, com 9%).

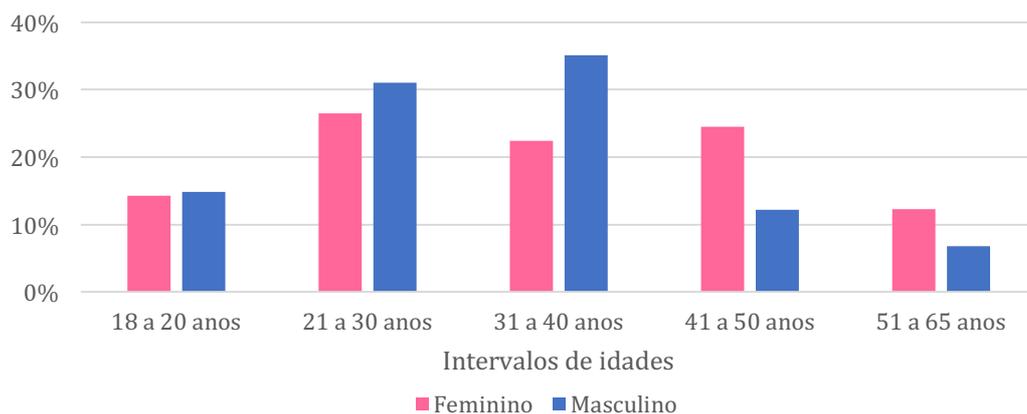


Imagem 16 – Cruzamento entre sexo e faixa etária

Da análise da Imagem 16 pode concluir-se que entre os 18 e os 30 anos ambos os sexos se distribuem nas faixas etárias de forma semelhante. De referir, ainda, a maior distribuição feminina nas faixas etárias mais velhas, com uma maior frequência das mulheres em faixas etárias superiores (41 anos ou mais) e menos na faixa etária central (31 a 40 anos).

Perfis de utilizador

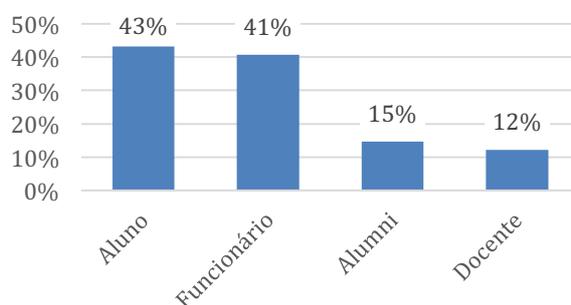


Imagem. 17 – Frequência de cada perfil de utilizador selecionado no inquérito

A Imagem. 17 mostra que os perfis de utilizador com percentagens mais elevadas são o de Aluno (43%) e de Funcionário não-docente (41%), seguindo-se o de Funcionário docente (15%) e de Alumni (12%). Assim, os principais utilizadores parecem ser os alunos bem como os funcionários não docentes da Universidade de Lisboa. Uma vez que o inquérito permitia aos participantes assinalar mais do que um perfil de utilizador em simultâneo, obtiveram-se categorias combinadas que, no entanto, apresentam percentagens baixas (entre 1% e 2%).

Instituição

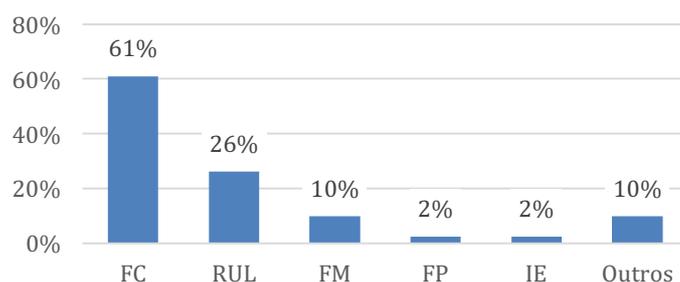


Imagem 18 – Frequência de cada instituição da ULisboa selecionada no inquérito

Legenda – FC: Faculdade de Ciências; RUL: Reitoria da ULisboa; FM: Faculdade de Medicina; FP: Faculdade de Psicologia; IE: Instituto de Educação.

A *Imagem 18* mostra que os participantes pertencem maioritariamente à Faculdade de Ciências (61%), seguindo-se a Reitoria (26%) e a Faculdade de Medicina (10%). Tal como no caso anterior, também aqui o questionário permitia escolher mais do que uma instituição, tendo 10% dos inquiridos respondido com mais do que uma instituição, sendo neste caso mais frequente a situação de pertença simultânea à Faculdade de Ciências e outra instituição ou à Reitoria e a outra instituição.

Métodos de aprendizagem preferidos

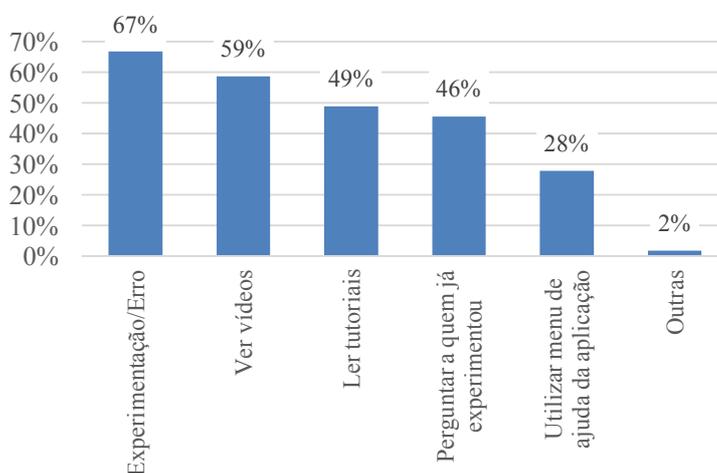


Imagem. 19 – Métodos de formação preferidos

De acordo com os dados da *Imagem. 19*, o método de formação mais escolhido pelos participantes no estudo é o de experimentação e erro (67%), o qual envolve a interação com a própria ferramenta, seguindo-se o da visualização de vídeos (59%). O

método menos assinalado pelos participantes é o da utilização de menus de ajuda da aplicação (28%). Dois participantes indicam outras opções (2%), respetivamente formação presencial e a frequência de *cursos online*. Tendo a oportunidade de utilizar vários métodos de formação em simultâneo, verifica-se que 14% dos participantes tendem a usar a combinação dos seguintes métodos de formação: ler tutoriais, ver vídeos e experimentação e erro.

Tempo médio de utilização de computador

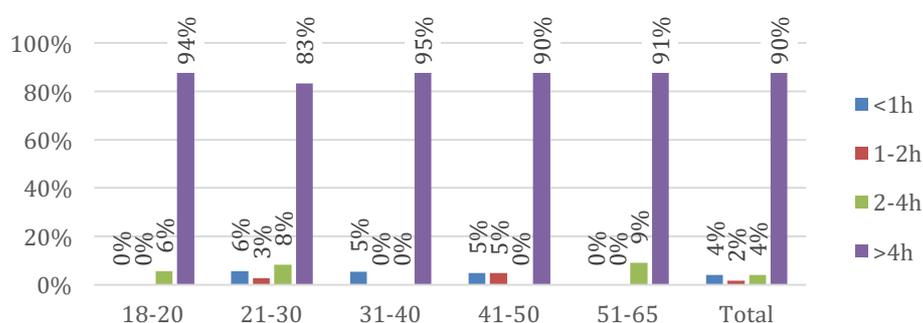


Imagem 20 – Tempo médio de utilização de computador por dia em relação à faixa etária

A *Imagem 20* mostra que a maioria dos inquiridos (90%) utiliza o computador mais de 4 horas por dia, em média. Apenas 4% utiliza o computador menos de uma hora por dia. Analisando o tempo médio de utilização diária de computador pelas cinco faixas etárias presentes no estudo, sublinha-se que para todas a maioria dos participantes faz uma utilização superior a 4 horas (variações entre 83% para os participantes com idades entre os 21 e os 30 anos e 95% nos participantes entre os 31 e os 40 anos). Adicionalmente, os mais jovens (18 a 20 anos) e os mais velhos (51 a 65 anos) utilizam diariamente o computador pelo menos 2 ou mais horas (não indicam períodos de tempo inferiores), ao passo que os participantes que se destacam com tempos de utilização mais reduzidos (menos de uma hora) são os da faixa etária entre os 21 e os 30 anos e os 31 e os 40 anos de idade (respetivamente, 6% e 5%).

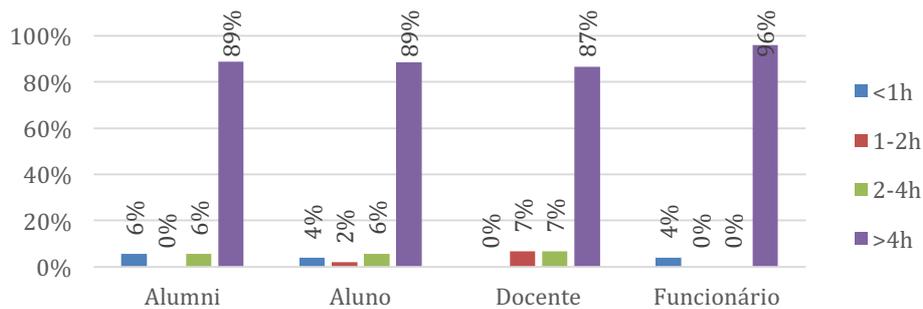


Imagem. 21 – Métodos de formação preferidos

A *Imagem. 21* apresenta os resultados do tempo médio de utilização de computador em relação aos perfis de utilizador. O tempo médio de utilização diária de computador por perfil de utilizador assume um comportamento similar ao anteriormente referido, ou seja, todos os perfis tendem a fazer uma utilização diária do computador superior a 4 horas (percentagens entre 87% e 96%).

Serviços de e-mail usados

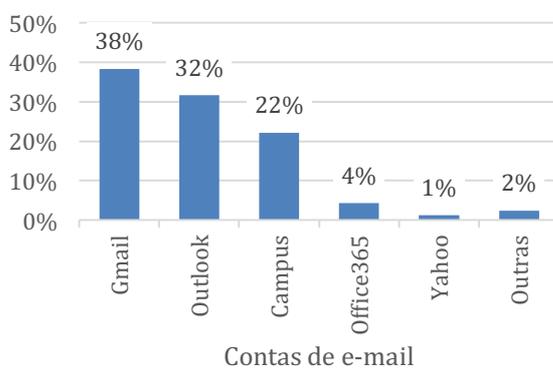


Imagem 22 – Contas de e-mail usadas regularmente

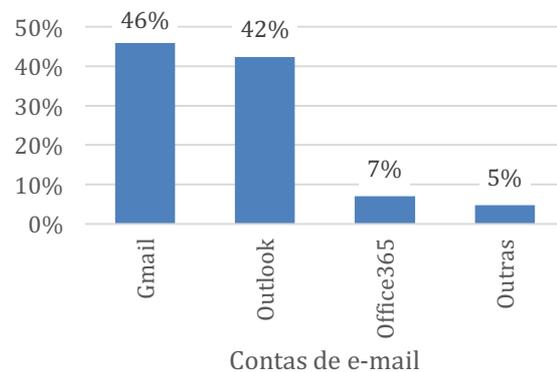


Imagem. 23 – Outras contas de e-mail dos utilizadores da conta Campus

Analisando a *Imagem 22*, verifica-se que os utilizadores utilizam maioritariamente a conta Gmail e Outlook, num total conjunto de 70%. A *Imagem. 23* mostra os serviços de e-mail usados regularmente pelos participantes que utilizam simultaneamente o e-mail Campus. Verifica-se que estes utilizadores seguem a mesma tendência geral indicada na *Imagem 22*, ou seja, usam preferencialmente a conta Gmail (46%) e a conta Outlook (42%). Outra conta utilizada com relativa expressão é a do Office365 (7%).

Ferramentas de TI usadas

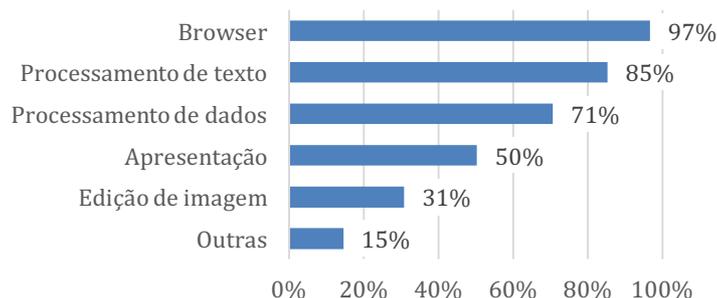


Imagem 24 – Ferramentas de TI utilizadas

A *Imagem 24* mostra que o *browser* de internet foi a ferramenta de TI mais escolhida pelos inquiridos (97%). As ferramentas de processamento de texto (por exemplo Word) também ficaram evidenciadas ao serem escolhidas por 85% da amostra, as ferramentas de processamento de dados (por exemplo Excel) com 71% e as ferramentas de preparação de apresentações (por exemplo PowerPoint) com 50%. As ferramentas de edição e manipulação de imagem tiveram uma percentagem de 31%.

Entre as restantes ferramentas indicadas pelos inquiridos, evidenciam-se as ferramentas de programação, editores de texto, IDEs (ambientes de desenvolvimento integrados) e ferramentas de comunicação (por exemplo Skype).

Frequência de alteração de senhas de acesso pessoais

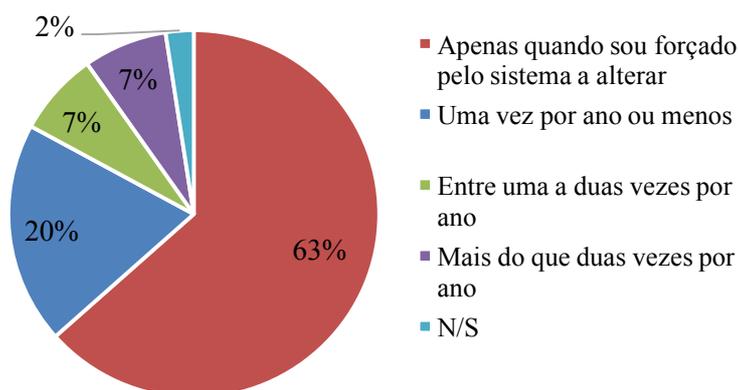


Imagem. 25 – Frequência da alteração da senha de acesso

Segundo os resultados apresentados na *Imagem. 25*, a maioria dos participantes apenas altera a senha quando é forçada pelo sistema a fazê-lo (63%), seguida daqueles que procedem a essa alteração uma vez por ano (20%).

Conhecimento acerca da conta Campus e seus serviços

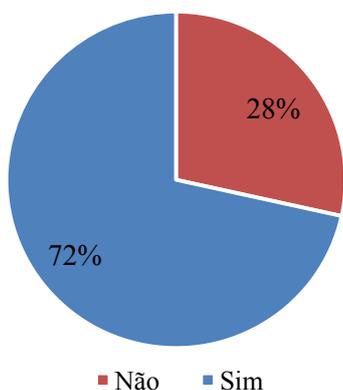


Imagem. 26 – Conhecimento da conta Campus da ULisboa e seus serviços

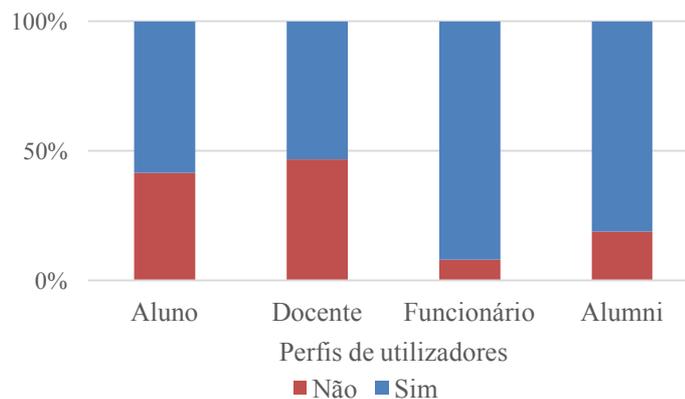


Imagem. 27 – Distribuição por perfil do conhecimento da conta Campus da ULisboa e seus serviços por perfil de utilizador

A maioria dos participantes conhece a conta Campus da Universidade de Lisboa e os seus serviços (72%), como ilustrado na *Imagem. 26*. Analisando o conhecimento das contas Campus e dos seus serviços por perfil de utilizador, tal como ilustrado na *Imagem. 27*, verifica-se que a distribuição é um pouco diferente entre os vários perfis de utilizador. São os funcionários não docentes quem mais conhece a conta Campus (92%), enquanto que alunos e docentes respondem com uma frequência muito menor que conhecem esta conta e os seus serviços (58% e 54% respetivamente).

Utilização da conta Campus

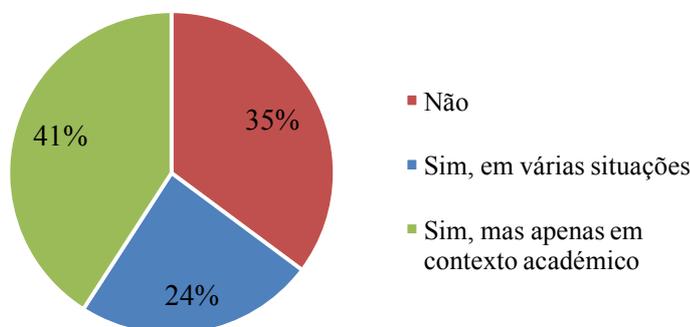


Imagem 28 – Utilização da conta Campus da ULisboa (n=88)

Dos 123 inquiridos, 88/72% respondeu afirmativamente sobre conhecer a conta Campus e os serviços associados (ver *Imagem. 26*). Destes 88, cerca de um terço (35%) refere que não utiliza a conta Campus, a maioria indica que apenas o faz em contexto académico (41%), e os restantes (24%) utilizam esta conta em várias situações, como mostra a *Imagem 28*.

Alteração regular da senha de acesso à conta Campus

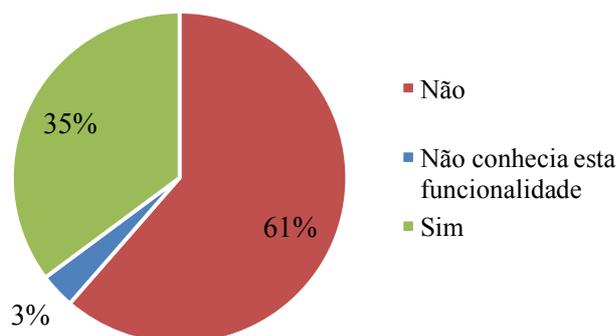


Imagem 29 – Alteração regular da senha de acesso da conta Campus da ULisboa (n=88)

Relativamente à alteração regular da senha de acesso da conta Campus, como mostra a *Imagem 29*, a maioria dos participantes indica não alterar (61%) ou não conhecer esta funcionalidade (3%). Apenas cerca de um terço dos utilizadores (35%) altera com regularidade a senha de acesso desta conta. Compare-se estes resultados com os apresentados na *Imagem. 25*: aproximadamente a mesma percentagem de pessoas que responde que não muda a senha da conta Campus indica também que só muda as senhas de acesso quando forçado pelo sistema. De facto, 74% dos inquiridos que responderam não mudar a senha de acesso da conta Campus indica só alterar senhas de acesso quando forçado; esta percentagem baixa para 48% entre os inquiridos que respondem alterar com regularidade a senha da conta Campus.

Local e dispositivo onde é usada a conta Campus

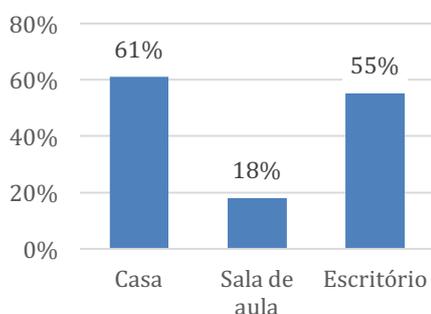


Imagem. 30 – Local onde os inquiridos usam o Portal de Utilizador (n=88)

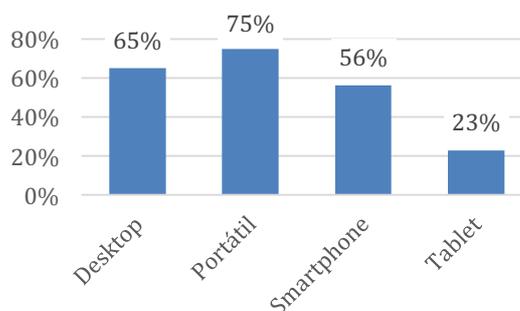


Imagem. 31 – Dispositivo usado pelos inquiridos para aceder ao Portal de Utilizador (n=88)

A *Imagem. 30* e a *Imagem. 31* mostram a distribuição dos locais onde os inquiridos usam o PU e os dispositivos que utilizados para o efeito. É importante salientar que em ambas as perguntas os inquiridos podiam escolher mais do que uma hipótese. A maioria dos utilizadores usa o Portal do Utilizador em casa (61%), sendo a sala de aula o local menos usado (18%). Já em relação aos dispositivos mais usados, tanto o *desktop* como o portátil e o *smartphone* são largamente usados (mais de 50% em qualquer um dos casos), sendo a quota de utilizadores que usa o *tablet* mais baixa (23%)

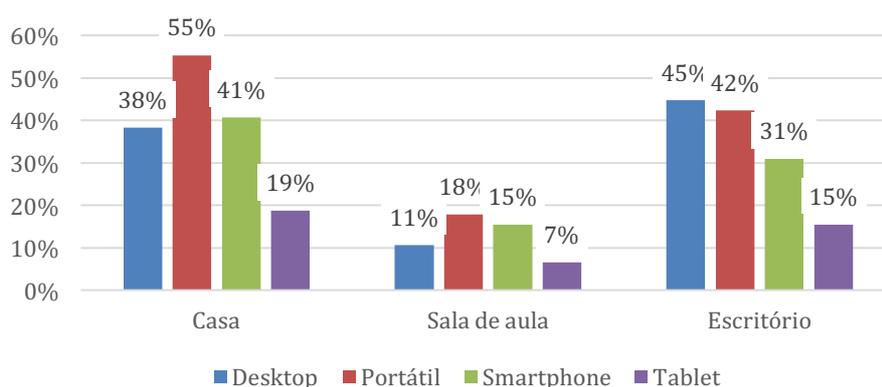


Imagem 32 – Local e Dispositivo de utilização do Portal de Utilizador

A *Imagem 32* mostra o cruzamento das respostas às duas perguntas anteriores. É relevante salientar que o dispositivo mais usado em casa ou em sala de aula é o portátil (55% e 18%, respetivamente), sendo que no escritório é o *desktop* o dispositivo mais comum (45%). A combinação mais comum é o uso de computador portátil em casa (55%). Outras combinações relevantes são o uso de dispositivos móveis (smartphone/15% e tablet/7%) na sala de aula, que é um ambiente por norma menos tranquilo e menos cómodo: apenas um total de 22% dos inquiridos faz uso de um destes dispositivos neste local.

Funcionalidades desejadas no PU

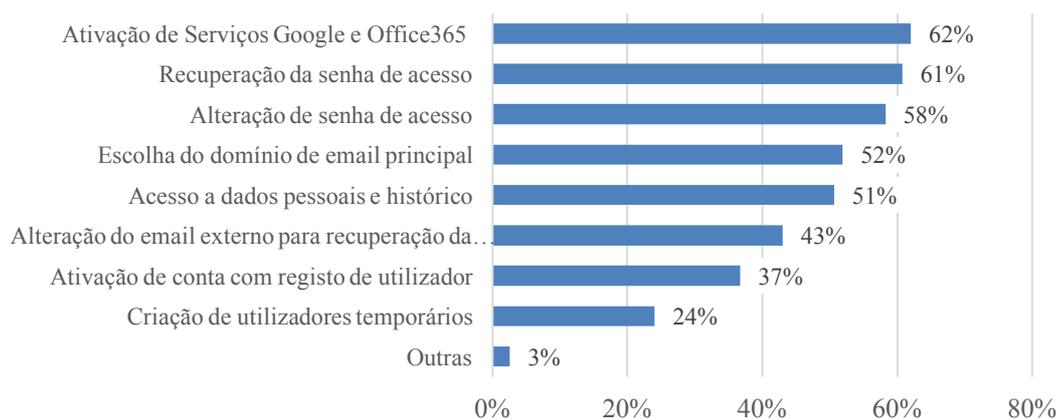


Imagem 33 – Funcionalidades desejadas selecionadas pelos inquiridos (n=79).

Sobre as funcionalidades desejadas para o novo Portal do Utilizador, ilustradas na *Imagem 33*, os inquiridos tendem a identificar “ativação de serviços Google e Office365 para utilizadores das Escolas da ULisboa que não utilizam a conta Campus com conta institucional” (62%), “recuperação da senha de acesso” (61%) e “alteração de senha de acesso” (58%). De entre as opções disponíveis, aquela que foi menos indicada pelos participantes foi a criação de utilizadores temporários (24%), provavelmente devido ao facto de ser uma funcionalidade apenas disponível para certos tipos de utilizador. Apenas dois participantes indicam outras funcionalidades, nomeadamente: “edição de dados pessoais” e “edição de dados de acesso aos diferentes serviços da ULisboa” e “recarregamento do cartão de aluno para os serviços do SAS”.

Refira-se ainda, relativamente à funcionalidade “Criação de utilizadores temporários”, que na questão “Quão frequente é criar utilizadores temporários” apenas 1 pessoa indica “uma vez por ano ou menos” e 4 respondem que não sabem.

4.1.2 Interpretação dos resultados

O mundo que nos rodeia será mais facilmente compreendido se poder ser quantificado (Graça Martins, 2005). Em todas as áreas do conhecimento é necessário saber “o que medir” e “como medir”. A Estatística é a ciência que ensina a recolher dados válidos, assim como a interpretá-los.

Pegando neste pressuposto, para uma população superior a 50 000 foi recolhida uma amostra de 123 inquéritos válidos, foram objeto de análise e tratamento. Refira-se,

contudo, que a dimensão da amostra de participantes face ao universo implica alguma reserva em termos de generalização das conclusões.

Das respostas obtidas, tecem-se as seguintes considerações finais:

- A maioria dos participantes no inquérito tem uma idade inferior a 41 anos e são funcionários não-docentes ou alunos, distribuindo-se de forma mais ou menos igualitária entre masculino e feminino.
- Relativamente ao método de formação preferencial com que aprendem a realizar as tarefas com ferramentas de TI no dia a dia, a maioria dos participantes referiu a técnica de experimentação e erro, visualização de vídeos e leitura de tutoriais.
- A grande maioria (quase absoluta), tem conhecimento de ferramentas de utilização da internet como o *browser*, processamento de texto e processamento de dados.
- Uma maioria esmagadora, independentemente do perfil (aluno, docente, funcionário), quer do sexo feminino ou masculino, utiliza o computador mais de 4 horas por dia e, dessa forma, pode supor-se que o universo ULisboa tem uma cultura em tecnologias de informação bastante elevada.
- A maior parte dos utilizadores, tenha ou não conta campus, possui mais do que uma conta de e-mail.
- Mais de dois terços dos utilizadores inquiridos têm conhecimento da conta Campus e conhece os serviços que são disponibilizados por utilizar essa conta.
- As funcionalidades/melhorias pedidas pelos utilizadores no novo Portal são: “recuperação da senha de acesso”, “ativação de serviços Google e Office365 para utilizadores que não utilizam conta Campus como institucional na Escola/Unidade a que têm um perfil associado”.
- Quanto ao dispositivo e o local onde irão aceder ao Portal para efetuar as tarefas pretendidas, infere-se que existe uma heterogeneidade de dispositivo a partir do qual o utilizador vai aceder (desktop, portátil, smartphone, tablet) e do local utilizado para esse efeito (casa, escritório, sala de aula).
- Apesar da literacia em TI dos utilizadores, o PU deve ser acessível a todo o universo da ULisboa e deverá ser fácil de usar, inclusivamente por pessoas que usam PC menos de uma hora por dia. É necessário a criação de mecanismos simples de interação com os utilizadores menos literatos em sistemas

informáticos do que o típico utilizador de sistemas e ferramentas de TI. E, por fim, é necessário ajustar os modelos conceptuais subjacentes à interface para que sejam acessíveis a estas pessoas.

- A maioria dos utilizadores altera apenas a senha de acesso quando o sistema obriga, e raramente altera a senha de acesso à conta Campus em particular.

4.1.3 Análise de tarefas

O desenho dos processos de *software* descreve como é que o modelo de negócio funciona, ou mais especificamente, descreve as atividades e as tarefas dos utilizadores na aplicação (Dufresne & Martin, 2003).

Um único processo pode consistir em muitos atores (pessoas, organizações ou sistemas) executando diversas tarefas. Para realizar uma tarefa global, os atores devem completar subtarefas específicas de forma coordenada. A maioria dos processos possui pontos de decisão onde o fluxo de processo se pode ramificar dependendo da condição do sistema ou da execução do processo em particular. Um processo pode ser descrito de forma diferente do ponto de vista dos diferentes atores. Uma metodologia de modelação de processos de *software* precisa de ser capaz de representar diferentes aspetos na descrição dos processos de *software*. Foi nesta fase do projeto que foi tida em conta a padronização dos processos de *software* e a implementação das regras de negócio de forma a permear a operação e os processos.

Para descrever as atividades e as tarefas dos utilizadores que irão interagir com a aplicação, recorreu-se a desenho de casos de uso. Estes casos de uso são apoiados na recolha de informação, nas conclusões dos resultados do inquérito em 4.1.2 , e das funcionalidades já existentes.

Um caso de uso é uma descrição em prosa do comportamento de um sistema ao interagir com o mundo exterior, representando uma lista de ações ou etapas de eventos que definem as interações entre o ator e um sistema para atingir um objetivo (Cockburn, 2002). Um caso de uso é uma metodologia utilizada na análise do sistema para identificar, esclarecer e organizar os requisitos do sistema. O caso de uso é constituído por um conjunto de possíveis sequências de interações entre sistemas e utilizadores num ambiente particular e que se relacionam com um objetivo específico. O caso de uso deve conter todas as atividades do sistema que tenham significado para os utilizadores (Rouse , 2007).

A análise de caso de uso é uma técnica de análise de requisitos importante e valiosa que tem sido amplamente utilizada na moderna engenharia de *software* desde a sua introdução formal por Ivar Jacobson em 1992.

Neste capítulo estão descritos os requisitos levantados para o Portal do Utilizador (PU). Os requisitos foram enumerados com base na informação obtida. O objetivo de listar os requisitos é a validação para depois serem definidos os casos de uso que ilustraram estes mesmos requisitos. Todos os requisitos são identificados por uma referência única que permite posteriormente avaliar o grau de cumprimento de cada requisito, e utilizar essa codificação para os testes que foram efetuados.

Descrição dos atores

- **Utilizador Criador (USR-CRT)**

O administrador é um utilizador com privilégios para criar utilizadores no repositório temporal para uma unidade/instituição específica (ou várias). O diagrama de casos de uso é representado pela *Imagem 35*. Os requisitos levantados para o ator Utilizador Criador (USR-CRT) foram:

- ✓ **USR-CR-01** – O Utilizador Criador pode alterar a senha de acesso
- ✓ **USR-CR-02** – O Utilizador Criador pode recuperar a senha de acesso
- ✓ **USR-CR-03** – O Utilizador Criador pode alterar o e-mail externo para recuperação de senha de acesso
- ✓ **USR-CR-04** – O Utilizador Criador pode efetuar um restabelecer da senha de acesso
- ✓ **USR-CR-05** – O Utilizador Criador pode aceder a dados pessoais
- ✓ **USR-CR-06** – O Utilizador Criador pode alterar o domínio principal da sua conta
- ✓ **USR-CR-07** – O Utilizador pode criar utilizadores temporários

- **Super-Administrador (SA-USR)**

O Utilizador Super-Administrador é um utilizador com privilégios para consulta de *logs*, estatísticas de acesso e parametrizações de sistema. Os requisitos levantados para o ator Super-Administrador (SA-USR) foram:

- ✓ **SA-USR-01** – O Super-Administrador pode alterar a senha de acesso
- ✓ **SA-USR-02** – O Super-Administrador pode recuperar a senha de acesso

- ✓ **SA-USR-03** – O Super-Administrador pode alterar o e-mail externo para recuperação de senha de acesso
- ✓ **SA-USR-04** – O Super-Administrador pode despoletar o processo de recuperação de senha de acesso
- ✓ **SA-USR-05** – O Super-Administrador pode aceder a dados pessoais
- ✓ **SA-USR-06** – O Super-Administrador pode alterar o domínio principal da sua conta
- ✓ **AS-USR-07** – O Super-Administrador pode efetuar parametrizações no Portal de Utilizador
- ✓ **AS-USR-08** – O Super-Administrador pode consultar através da aplicação, os *logs* que estão a ser registados
- ✓ **AS-USR-09** – O Super-Administrador pode consultar estatísticas de acesso relativas ao Portal do Utilizador

- **Utilizador (USR)**

Considera-se utilizador, todo aquele que interage com aplicação e efetua operações com sucesso.

O utilizador pode pertencer a diferentes repositórios (Temporal, Users, Pending e External), consoante os atributos que tem preenchidos. A restrição de acesso ao tipo de operações que pode efetuar no portal, é determinada pelo repositório a que pertence, conseqüentemente pelo valor dos atributos que tem preenchidos. A *Imagem 6* apresenta o circuito de alteração de repositório dos utilizadores, consoante a alteração do preenchimento dos atributos. O diagrama de casos de uso é representado pela *Imagem 36*.

Os requisitos levantados para a figura utilizador (USR) foram:

- ✓ **USR-01** – O Utilizador pode alterar a senha de acesso
- ✓ **USR-02** – O Utilizador pode recuperar a senha de acesso
- ✓ **USR-03** – O Utilizador pode alterar o e-mail externo para recuperação de senha de acesso
- ✓ **USR-04** – O Utilizador pode despoletar o processo de recuperação da senha de acesso
- ✓ **USR-05** – O Utilizador pode ativar a conta
- ✓ **USR-06** – O Utilizador pode aceder a dados pessoais

- ✓ **USR-07** – O Utilizador pode ativar a conta para aceder aos serviços Google for *Education* e Office365
- ✓ **USR-08** – O Utilizador pode alterar o domínio principal da sua conta

Diagramas dos Casos de Uso

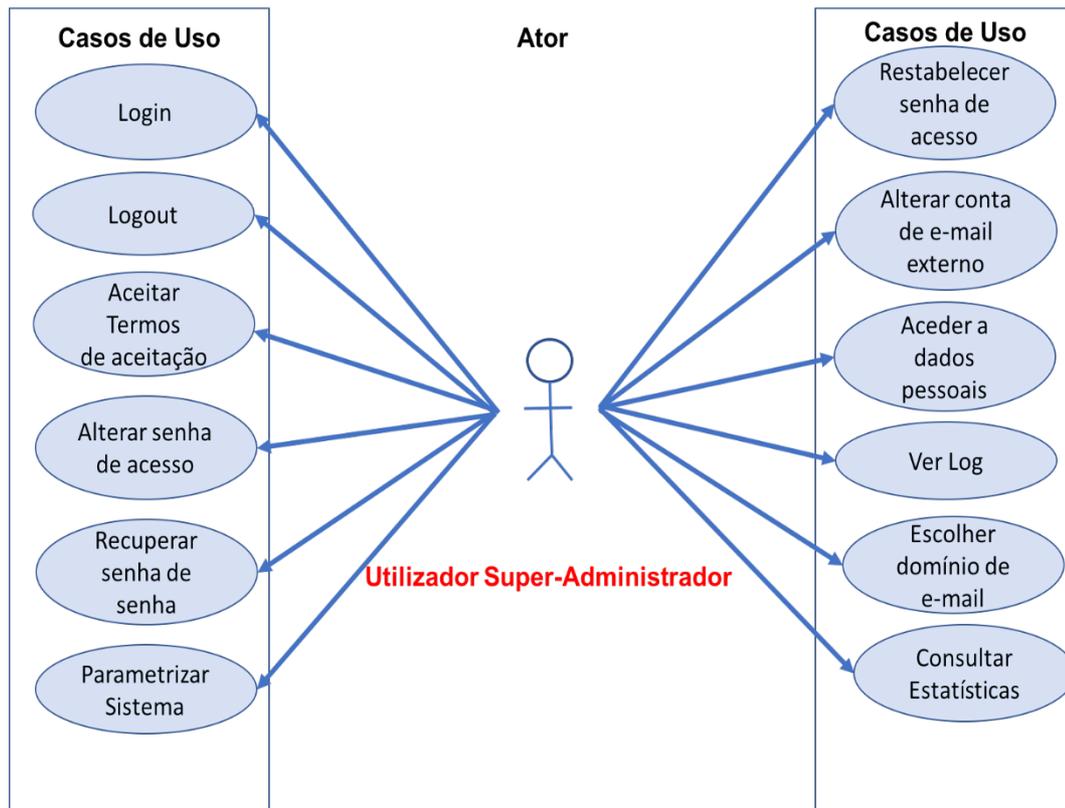


Imagem 34- Diagrama de Casos de uso do Super Administrador (SA-USR)

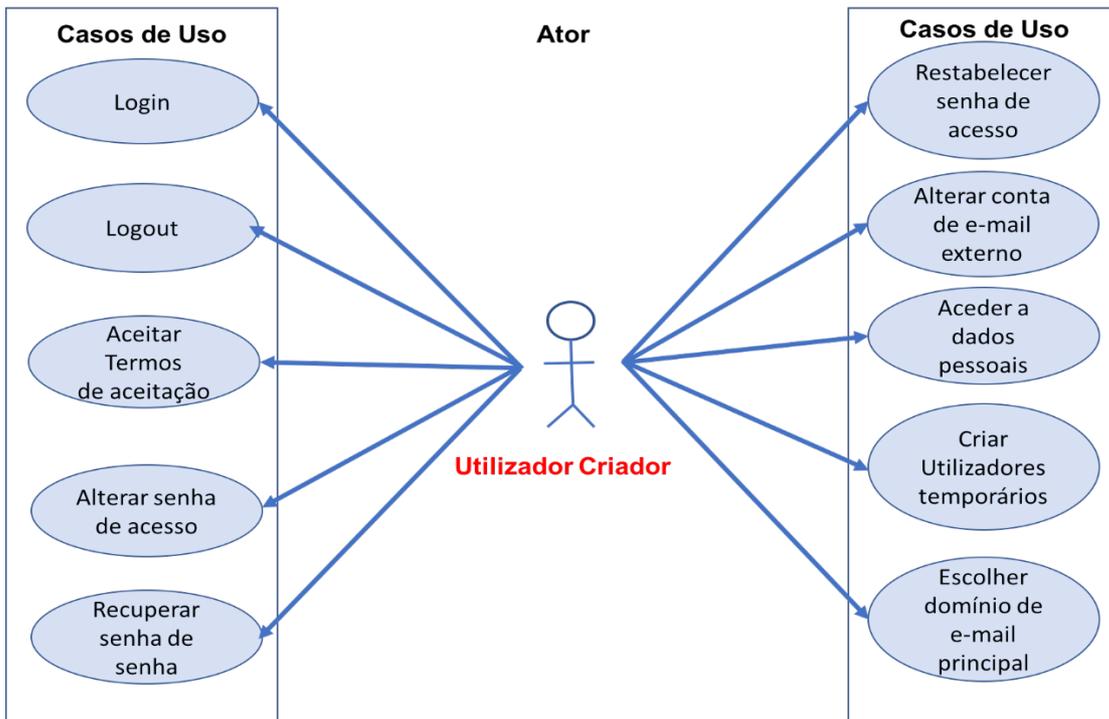


Imagem 35 – Diagram de Casos de uso do Utilizador Criador (USR-CR)

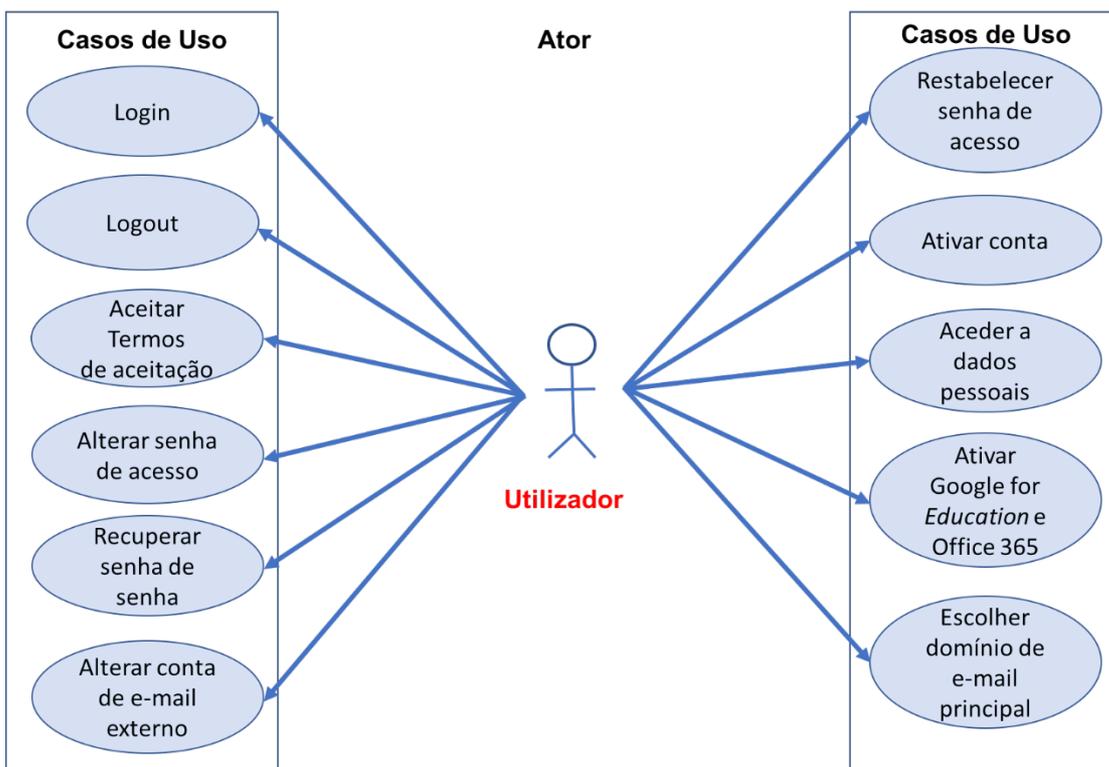


Imagem 36 - Diagrama de casos de uso do Utilizador (USR)

Eventos do IdM

Os eventos do IdM, com um conjunto de instruções definidas, são despoletados quando ocorrem alterações de valores em determinados atributos pré-estabelecidos. Para se efetuar o desenho e implementação de um evento no IdM, é necessário definir quais são os *inputs* para que atributo (ou atributos) despoletem o evento, e qual o *output* esperado após a execução da rotina.

Nesta fase foram definidos os *inputs* necessários que o PU tem de manipular, para enviar para o eDirectory do IdM.

Foi também levantado um conjunto de eventos necessários para que o Portal do Utilizador possa interagir com o IdM, consoante os casos de uso especificados, e poder despoletar as ações necessárias para as funcionalidades que foram implementadas. As ações nos conetores do IdM são despoletadas, consoante os *inputs* que o PU envia para o eDirectory do IdM.

4.2 Desenho técnico e funcional

Anteriormente à codificação, existe uma etapa preponderante, onde é garantida a arquitetura da aplicação, a estrutura de dados, os servidores, e como são definidas as ligações entre os diversos componentes de *software*.

Podemos definir *Software* como uma aplicação que gere e suporta os recursos e operações de um computador enquanto executa várias tarefas.

Estrutura de dados

Para que a informação do utilizador seja apresentada, tratada e manipulada, foi necessário efetuar um levantamento dos atributos do eDirectory necessários. Esse levantamento está refletido nos anexos Anexo I, Anexo II e Anexo III.

As estruturas de dados foram desenhadas com base nos atributos do eDirectory levantados, para poderem ser manipulados pela aplicação. As entidades java (EJB) criadas, representam as tabelas da base de dados relacional Oracle.

Arquitetura do *software*

A arquitetura de *software* recebeu atenção crescente como um importante subcampo de engenharia de *software*. A definição de uma arquitetura de *software*

correta é um fator de sucesso para o design e desenvolvimento de sistemas. Uma boa arquitetura de *software* pode ajudar a garantir que um sistema satisfaça os requisitos essenciais em áreas como desempenho, confiabilidade, portabilidade, escalabilidade e interoperabilidade. No entanto, se a arquitetura não for bem definida, pode ser ineficiente, e em último recurso desastrosa para o projeto.

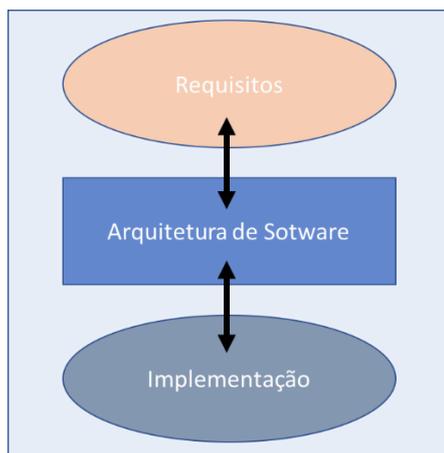


Imagem 37 – Arquitetura de *software* como uma ponte entre os requisitos e a Implementação adaptado de (Garlan, 2000)

Existem inúmeras definições de arquitetura de *software*, mas todas concordam que a arquitetura descreve um sistema na forma bruta, e mostra o nível mais elevado do comportamento de todos os componentes que vão interagir uns com os outros. Pode dizer-se também que a arquitetura de *software* desempenha o papel de ponte entre os requisitos e a implementação, representado na *Imagem 37*, pois será a partir dos requisitos que se elaborará a arquitetura que depois será implementada em código (Garlan, 2000).

O *software* que está a ser construído nos dias de hoje tem vindo a ganhar complexidade. A melhor forma de reduzir tempos, custos de desenvolvimento e manutenção, é recorrer-se a um desenho de arquitetura que permita avaliar a complexidade do sistema que está a ser desenvolvido, para os qual foram levantados requisitos.

Na *Imagem 38* está representada a arquitetura de Software do Portal do Utilizador, com os vários componentes e ligações que existem entre si. A aplicação desenvolvida, é uma aplicação web, que poderá ser acedida de a partir de qualquer dispositivo com

programas habilitados a interagem com documentos HTML, javascript e CSS, através da rede (*browser* de internet)

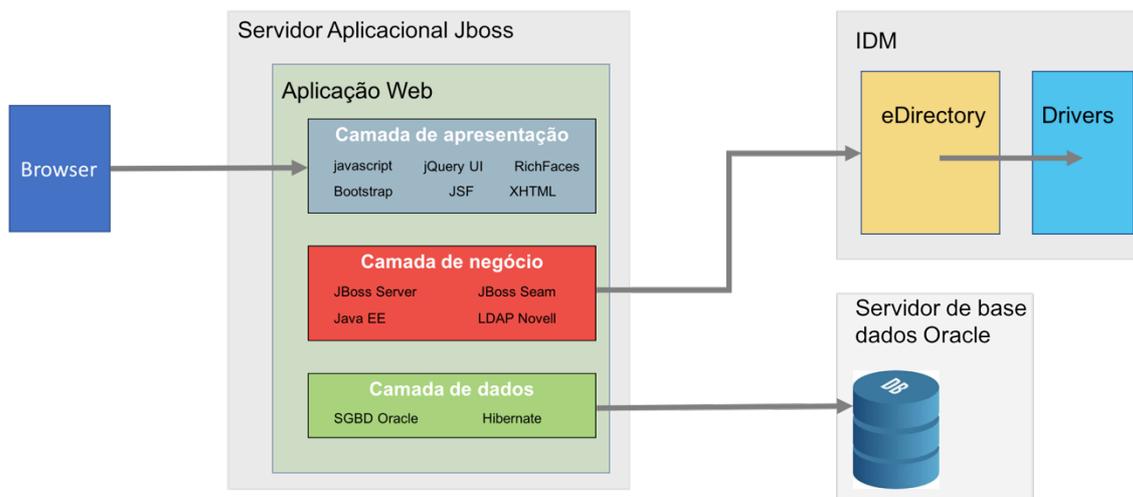


Imagem 38 - Arquitetura de *software* do Portal do Utilizador

Os módulos da aplicação, sobre o formato EAR, correm através do servidor aplicacional JBoss. As diferentes camadas da aplicação são colocadas na instância do servidor aplicacional JBoss, na pasta de *deploy*, no formato EAR. É a camada de negócio a responsável pela comunicação entre as camadas de apresentação e camada de dados. É nesta mesma camada também, através da API da Novell, que a aplicação efetua as chamadas ao eDirectory, que consoante o preenchimento dos atributos, despoleta as diferentes ações nos *conectores* do IdM.

Caracterização das interfaces

As interfaces são componentes essenciais num sistema de *software*, pois é através delas que o utilizador comunica com o sistema, para efetuar as tarefas. Uma interface de *software* estabelece uma interação pessoa-máquina em ambos os sentidos, pois o utilizador estabelece um “diálogo” com a máquina, e esta responde consoante o *input* enviado permitindo um *feedback* da ação.

Hoje em dia as interfaces tendem a ser o mais amigáveis possíveis, e independentes do sistema operativo, ou dispositivo que são acedidas, utilizando para isso tecnologias responsivas que se adaptam aos ambientes onde estão a ser executadas. Para que seja garantida esta capacidade responsiva ao sistema operativo e ao dispositivo, os programadores recorrem a tecnologias como o Bootstrap (ver 3.2.4 , que

permitem a geração de componentes capazes de ser interpretados em ambientes mais díspares. Na implementação do Portal do Utilizador, para tornar a aplicação responsiva foram utilizadas tecnologias Bootstrap, apoiadas por RichFaces e jQuery.

A cada caso de uso descrito, vai estar associado um esboço de interface que corresponderá ao cenário que o caso de uso descreve. Os protótipos das interfaces são desenhados após o levantamento de requisitos e descrição dos diversos casos de uso.

No projeto de implementação do Portal de Utilizador foi desenhado para caso de uso, um esboço de uma interface. Os esboços (também denominados por protótipos de baixa fidelidade, ou *wireframes*) estão disponibilizados no Anexo VII. A componente criativa de criação do desenho a partir dos protótipos de interfaces esteve a cargo do Núcleo de Comunicação dos Serviços Centrais da Universidade de Lisboa.

Para exemplificar um caso de uso com a ligação ao protótipo de interface, recorri ao caso de uso de alteração da senha de acesso apresentado na *Imagem 39*. Note-se que todos os casos de uso estão descritos de forma semelhante no Anexo VI.

- Caso de Uso: **ALTERAR SENHA DE ACESSO**

Atores: USR, USR-CR, SA-USR

Pré-condições:

- O ator tem de estar autenticado.
- A senha de acesso tem de obedecer às regras indicadas em Anexo V

Pós-condições:

- A senha é alterada com sucesso

Descrição:

- 1.1- O ator escolhe “ALTERAR SENHA”
- 1.2- O ator insere a sua senha atual
- 1.3- O ator insere a nova senha
- 1.4- O ator repete a nova senha
- 1.5- O ator confirma pressionando “ALTERAR SENHA”.



Imagem 39 – Protótipo de interface da página de alterar senha de acesso

Neste protótipo de interface são descritos os passos que o utilizador tem de efetuar para poder alterar a senha de acesso com sucesso.

4.3 Implementação

É nesta fase que todo o trabalho técnico e funcional é traduzido para código, ou seja, implementado. Foram incluídos testes unitários nos módulos desta etapa para que as unidades de código produzidas serem testadas, como recomenda as especificações do modelo sequencial linear na *Imagem 8*. Para a criação destes testes unitários, recorreu-se ao *framework* TestNG descrita em 3.2.12 .

Para a implementação do Portal de utilizador, foram utilizadas as mesmas tecnologias utilizadas pelo NDS/DI para desenvolvimentos internos, de forma a encurtar o período de implementação, reduzir o custo de aprendizagem, e aumentar a capacidade de suporte tecnológica, caso seja necessário.

Para a implementação da componente gráfica foram utilizadas tecnologias Bootstrap, referida em 3.2.4 ., apoiadas por jQuery e Richfaces, como referido no subcapítulo de Caracterização das interfaces.

4.4 Testes

Após a implementação do desenho de interface e de toda a codificação que teve como base os requisitos levantados, foram efetuados controlos de qualidade do *software* desenvolvido. Desta forma foi assegurado no projeto uma “auditoria” a todas as funcionalidades implementadas e foi assegurado que as mesmas funcionam corretamente. É nesta etapa do projeto que é feito o controle de qualidade.

Segundo o *standard* ANSI/IEEE 1059, testar pode ser definido como um processo para analisar *software* para detetar as condições existentes e exigidas (erros, defeitos e bugs), e avaliar os recursos da peça de *software*. O objetivo de testar é verificar, validar e detetar erros para procurar problemas, para os mesmos poderem ser corrigidos (Isha, 2014). Um bom teste de *software* é aquele que tem uma grande probabilidade de encontrar um erro que ainda não foi descoberto.

Existem diversas técnicas de teste. As mais utilizadas são os testes de caixa preta e testes de caixa branca. O primeiro testa o *software* (código) e o segundo a componente de interface (testes funcionais).

Os testes permitiram uma validação das funcionalidades, layout e usabilidade. Foram feitos também testes de carga utilizando ferramentas para esse efeito, de forma a permitir um comportamento idêntico às sobrecargas esperadas num ambiente de produção.

Para a elaboração dos testes, recorreu-se a *software* que simula os dois tipos de técnicas de teste: testes de caixa branca e testes de caixa preta. O TesteNG descrito em 3.2.12, permite a elaboração de uma bateria de testes que agrega os dois modelos de teste mencionados, conseguindo simular o utilizador e a interface com diferentes *inputs*, testes de carga ao servidor, e testes unitários para despistagem de erros.

4.5 Manutenção

A manutenção de *software* é o processo de melhoria e otimização de um *software* já desenvolvido. A manutenção do *software* é uma das fases do processo de desenvolvimento de *software*, e ocorre após a entrada do *software* em produção. (Wikipedia, 2018).

No entanto a manutenção de *software* não é consensual, pois podem ser considerados diferentes tipos de manutenção de *software*.

A manutenção corretiva é composta pela correção de erros no *software* que foram detetados na fase de testes. A manutenção adaptativa inclui adaptações do *software* a ambientes externos e regras de negócio. A manutenção evolutiva é um tipo de manutenção que visa adicionar funcionalidades ao *software* existente, proporcionando-lhe um aumento de funcionalidades e aumento de desempenho (Leandro MTR, 2015). Para existe mais uma categoria a considerar, a manutenção preventiva. Este tipo de manutenção considera fatores como o aumento de desempenho da aplicação e a prevenção de eventuais anomalias que poderão existir no sistema (Pressman & Maxim, 2014).

Capítulo 5

Discussão e conclusão

Com a necessidade e vontade por parte do AASI/DI/SC de disponibilizar um melhor serviço ao utilizador da ULisboa, com maiores funcionalidades, cuja antiga página web do utilizador já não conseguia suportar, surgiu o projeto para elaboração do Portal de Gestão do Utilizador da ULisboa. Com início do projeto de *upgrade* ao sistema IdM instalado e do projeto de implementação de autenticação federada na ULisboa, esta necessidade tornou-se prioritária.

Este Portal mantém as funcionalidades existentes na antiga página e acrescenta novas. São exemplos de novas funcionalidades pretendidas a recuperação da senha de acesso, ativação da conta *Google for Education* e a escolha do domínio principal do utilizador. A antiga página do utilizador já permitia a criação/registo de utilizadores, e a alteração da senha de acesso.

Todas as funcionalidades foram redesenhadas e foi elaborada uma refatorização de código completa para colmatar as ineficiências que existiam na antiga página dos utilizadores.

O objetivo principal desta tese de projeto de mestrado é a reimplementação das funcionalidades da antiga página de utilizadores, criação de novas funcionalidades, e permitir que o Portal do Utilizador tenha comportamentos responsivos e adaptativos nos ambientes onde for acedido, seja móvel ou *desktop*.

O novo Portal interage com a arquitetura de IdM, efetuando chamadas via LDAP, que consoante as operações de alteração de atributos, despoleta eventos do lado do IdM (por exemplo, a criação de um novo utilizador).

Para o desenvolvimento deste novo Portal serão utilizadas ferramentas e tecnologias de desenvolvimento já testadas e comprovadas em outros desenvolvimentos realizados pelo NDS/AASI/DI. A delicadeza do projeto, e o tempo curto de execução,

não deixou grande manobra para se investir em *framework* de desenvolvimento mais recentes. Todas as funcionalidades pretendidas conseguiram ser asseguradas pelas tecnologias até agora utilizadas pelo NDS, permitindo assim, neste componente tão importante, a depuração de eventuais variáveis indesejadas que possam ocorrer.

Este Portal do Utilizador trará para toda a comunidade Académica da ULisboa um Portal desenvolvido para o utilizador, com um acrescento de novas funcionalidades solicitadas pelos utilizadores, e que permitirá um alívio no suporte informático ao utilizador, com mecanismos de recuperação de senha de acesso, informações ao utilizador, ativação de conta Google e alteração do domínio principal de conta que o utilizador pretende utilizar.

Foi também objetivo deste projeto a elaboração de um conjunto de mecanismos para produção de relatórios estatísticos e que permita a avaliação do registo de eventos aplicativos do Portal do Utilizador (*log*). Para esse efeito, foram elaborados mecanismos de registos de todas as operações numa base de dados, permitindo depois o tratamento da informação recolhida. Este elemento permitirá também avaliar a quantidade de utilizadores que se encontram no Portal, para análise de carga, de forma a facilmente poderem ser despoletados mecanismos de alertas pelos colegas de suporte.

Este componente do projeto de gestão de identidade da ULisboa, o Portal do Utilizador, tem também como objetivo iniciar princípios que permitam estabelecer relações de confiança entre serviços em diferentes contextos, para uma modificação importante das tecnologias de Gestão de Identidade presentes na ULisboa, autorização e autenticação, permitindo uma gestão federada dos perfis de utilizadores e das políticas de acesso.

Referências Bibliográficas

- Ahn, L. V., Maurer, B., McMillen, C., Abraham, D., & Blum, M. (12 de 09 de 2008). reCAPTCHA: Human-Based Character Recognition via Web Security Measures. *Science Magazine*.
- Allen, D. (September 2008). *Seam In Action*. Manning Publications.
- Bertino, E., & Takahashi, K. (2011). *Identity Management: Concepts, Technologies, and Systems*. Artech House.
- Bishop, M. (2002). *Computer Security Arts and Science*. Addison-Wesley.
- Bootstrap - front-end framework*. (s.d.). Obtido em 2016, de Wikipédia: [https://en.wikipedia.org/wiki/Bootstrap_\(front-end_framework\)](https://en.wikipedia.org/wiki/Bootstrap_(front-end_framework))
- Bootstrap 3 Tutorial*. (s.d.). Obtido em 2016, de w3schools: <http://www.w3schools.com/bootstrap/>
- Briggs, S., & Spence, S. (2016). *LDAP*. Obtido de Search Mobile Computing: <http://searchmobilecomputing.techtarget.com/definition/LDAP>
- Carnegie Mellon University. (2010). *CAPTCHA: Telling Humans and Computers Apart Automatically*. Obtido em 23 de 09 de 2017, de Captcha: <http://www.captcha.net/>
- Cockburn, A. (Mar/Apr de 2002). *Use cases, ten years later*. Obtido em 23 de 09 de 2017, de Alistair Cockburn: <http://alistair.cockburn.us/Use+cases%2c+ten+years+later>
- Definição de Hibernate*. (s.d.). Obtido em 2017, de Wikipédia: [http://en.wikipedia.org/wiki/Hibernate_\(Java\)](http://en.wikipedia.org/wiki/Hibernate_(Java))
- Dufresne, T., & Martin, J. (2003). Process Modeling for E-Business. *INFS 770 Methods for Information Systems Engineering: Knowledge Management and E-Business Framework*. (s.d.). Obtido em 2016, de Wikipédia: <https://en.wikipedia.org/wiki/Framework>

- Flanagan, D., & Ferguson, P. (2002). *JavaScript: The Definitive Guide* (4th ed.). O'Reilly & Associates.
- Fonseca, M. J., Campos, P., & Gonçalves, D. (2012). *Introdução ao Design de Interfaces*. FCA.
- Garlan, D. (2000). Software Architecture: a Roadmap. *School of Computer Science at Research Showcase @ CMU*, (p. 06).
- Gartner. (2016). *Identity and Access Management (IAM)*. Obtido de Gartner: <http://blogs.gartner.com/it-glossary/identity-and-access-management-iam/>
- Graça Martins, M. E. (2005). *Introdução à Probabilidade e à Estatística - Com complementos de Excel*. Lisbon: Sociedade Portuguesa de Estatística.
- History of LDAP*. (2017). Obtido de <https://ldapwiki.com:https://ldapwiki.com/wiki/History%20of%20LDAP>
- Howes, T. A., Smith, M. C., & G, G. S. (2003). *Understanding and Deploying LDAP Directory Services*. Addison-Wesley Longman Publishing Co.
- Isha, S. S. (2014). Software Testing Techniques and Strategies. *Isha Int. Journal of Engineering Research and Applications*, 99-102.
- ITU, T. S. (January de 2009). *Y.2720 : NGN identity management framework*. Obtido em 2017, de <https://www.itu.int/rec/T-REC-Y.2720-200901-I:https://www.itu.int/rec/T-REC-Y.2720-200901-I>
- JSR 127: JavaServer Faces*. (2016). Obtido de Java Community Process: <https://www.jcp.org/en/jsr/detail?id=127>
- LDAP Classes for Java*. (s.d.). Obtido em 2016, de Novell: https://www.novell.com/developer/ndk/ldap_classes_for_java.html
- Leandro MTR. (13 de 04 de 2015). *Tipos de manutenção de Software*. Obtido de Leandro MTR: <http://www.leandromtr.com/tecnologia-informacao/tipos-de-manutencao-de-software/>
- Linguagens de Marcação*. (s.d.). Obtido de Wikipédia: https://en.wikipedia.org/wiki/Markup_language
- Microsoft. (2016). *What is LDAP*. Obtido de Microsoft Developer Network: [https://msdn.microsoft.com/en-us/library/aa367036\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/aa367036(v=vs.85).aspx)
- Nabeth, T. (2009). Identity of Identity. Em K. Rannenberg, D. Royer, & A. Deuker (Edits.), *The Future of Identity in the Information Society* (pp. 19-69). New York: Springer.

- O que é Java EE?* (2016). Obtido de NetBeans: https://netbeans.org/kb/trails/java-ee_pt_BR.html
- Oracle Help Center. (2016). *Comparison of the LDAP and JNDI Model*. Obtido de Oracle Help Center: <http://docs.oracle.com/javase/jndi/tutorial/ldap/models/v3.html>
- Pfitzmann, A., & Hansen, M. (10 de 09 de 2010). *Anonymity, Unlinkability, Undetectability, Unobservability, Pseudonymity, and Identity Management—A Consolidated Proposal for Terminology*. Obtido de Technische Universität Dresden: Anonymity, Unlinkability, Undetectability, Unobservability, Pseudonymity, and Identity Management—A Consolidated Proposal for Terminology
- Pressman, R. S., & Maxim, B. (2014). *Software Engineering: A Practitioner's Approach* (8th Edition ed.). McGraw-Hill Science/Engineering/Math.
- Reis, E., Melo, P., Andrade, R., & Calapez, T. (s.d.). *Estatística Aplicada - Probabilidades, Variáveis aleatórias, Distribuições Teóricas* (6 ed., Vol. 1). EDIÇÕES SÍLABO, LDA.
- RichFaces 3.3.X*. (s.d.). Obtido em 2016, de RichFaces Developer Guide: http://docs.jboss.org/richfaces/latest_3_3_X/en/devguide/html/
- Rouse, M. (07 de 2007). *Defenition use case*. Obtido em 23 de 09 de 2017, de Search Softwary Quality: <http://searchsoftwarequality.techtarget.com/definition/use-case>
- Royce, W. W. (26 de August de 1970). Managing the Development of Large Software Systems. *IEEE WESCON 26*, pp. 1–9.
- Seam: Repenser l'architecture des applications web?*. (s.d.). Obtido em 2017, de Xebia: <http://blog.xebia.fr/2009/06/03/seam-repenser-larchitecture-des-applications-web/>
- Searls, D. (31 de 12 de 2002). *Myidentity&Ouridentityvs.Theiridentity*. Obtido em 2017, de Doc Searls Weblog: <http://doc.weblogs.com/2002/12/31#myidentityOuridentityVsTheiridentity>
- Software, Gracion. (2016). *What is LDAP*. Obtido de Gracion Software: <http://www.gracion.com/server/whatldap.html>
- Techopedia. (s.d.). *Application Server*. Obtido em 23 de 09 de 2017, de Techopedia: <https://www.techopedia.com/definition/432/application-server>

- Torvalds, L. (10 de 09 de 2017). *Git*. Obtido de Wikipédia:
<https://en.wikipedia.org/wiki/Git>
- What is a "Facelet"?* (s.d.). Obtido em 2017, de JSF tool box:
<http://www.jsftoolbox.com/documentation/facelets/03-FaceletsConcepts/composite-view-design-pattern.jsf>
- Wikipedia. (25 de 09 de 2018). *Manutenção de software*. Obtido de Wikipédia:
https://pt.wikipedia.org/wiki/Manuten%C3%A7%C3%A3o_de_software
- Zeilenga, K. D. (2006). *Lightweight Directory Access Protocol (LDAP): Technical Specification Road Map*. Obtido de The Internet Engineering Task Force (IETF): <https://tools.ietf.org/html/rfc4510>

Glossário de Acrónimos

NDS – Núcleo de Desenvolvimento de Software

AASI – Área de Sistemas de Informação da Área de Sistemas de Informação

DI – Departamento de Informática

SC – Serviços Centrais

CSS – Folha de estilos em cascata (*Cascading Style Sheets*)

JS – JavaScript

XML - eXtensible Markup Language

HTML - HyperText Markup Language

XHTML - eXtensible HyperText Markup Language

APIs - Interfaces de programação de aplicações

Ajax - Asynchronous JavaScript and XML

JSF – Java Server Faces

Java EE - Java Enterprise Edition

Java SE – Java Standard Edition

JCA – Java Connector Architecture

EJB – Enterprise Java Beans

EAR – Enterprise Application Archive

JTA – Java Transaction API

SGBD – Sistema de Gestão de Base de Dados

IdM - Identity Management

PU – Portal de Utilizador

AM – Access Manager

IP – Internet Protocol

OO – Object Oriented

GCV- Global Configuration Value

USR – Utilizador do Portal do Utilizador

USR-CRT – Utilizador com permissões de criação de utilizadores

SA-USR – Utilizador super-administrador

Anexo I Valores dos sufixos de atributos

Lista de Escolas/Unidades a usar como valores ou sufixo de atributos:

Sigla	Nome da Escola/Unidade
EUL	Estádio Universitário de Lisboa
FA	Faculdade de Arquitectura
FBA	Faculdade de Belas-Artes
FC	Faculdade de Ciências
FD	Faculdade de Direito
FF	Faculdade de Farmácia
FL	Faculdade de Letras
FM	Faculdade de Medicina
FMD	Faculdade de Medicina Dentária
FMH	Faculdade de Motricidade Humana
FMV	Faculdade de Medicina Veterinária
FP	Faculdade de Psicologia
ICS	Instituto de Ciências Sociais
IE	Instituto de Educação
IGOT	Instituto de Geografia e Ordenamento do Território
ISA	Instituto Superior de Agronomia
ISCSP	Instituto Superior de Ciências Sociais e Políticas
ISEG	Instituto Superior de Economia e Gestão
IST	Instituto Superior Técnico
RUL	Reitoria da Universidade de Lisboa
SAS	Serviços de Ação Social

Anexo II Atributos da classe ULAuxUser

Nome Atributo	Tipo	Origem	Observações
eduPersonPrimaryAffiliation	String		Atributo usado para serviços federados da FCCN
FullName	String	SAP RH, SIGES, GIAF, Fénix, Portal Utilizador	Nome completo
GivenName	String	SAP RH, SIGES, GIAF, Fénix, Portal Utilizador	Nome próprio (por default, este atributo vem com limite 32 caracteres)
ULBI	String	SAP RH, SIGES, GIAF, Fénix	Documento de identificação
ULBirthDate	String	SAP RH, SIGES, GIAF, Fénix	Data Nascimento
ULCountryCode	String	SAP RH, SIGES, GIAF, Fénix	País de Origem
ULExternalEmailAddr	String	SAP RH, SIGES, GIAF, Fénix, Portal Utilizador	Endereço de e-mail pessoal
ULFaculdades	String		Atributo multi-valor com todas as SIGLAS das Faculdades que o utilizador pertence/pertenceu.
ULInternalUser	Boolean		Indica se a conta do utilizador foi criada via Sistema BD interno.
ULGivenName	String	SAP RH, SIGES, GIAF, Fénix, Portal Utilizador	Nomes próprios do utilizador (foi criado este atributo pela necessidade de conter todos os nomes próprios do utilizador, pois o atributo default do LDAP apenas permite 31 caracteres).
ULFacRolesTm	Date		Atributo criado com a finalidade de despoletar determinados tipos de eventos. (Ex: If ULFacRolesTm changing, Then < Actions>)
ULisOfExternalFac	Boolean		Atributo utilizado para vetar a sincronização de utilizadores de algumas faculdades para alguns serviços

ULGoogleAppsAccountRequest	String		
ULGoogleAppsMain	String		Domínio principal escolhido pelo utilizador
ULGoogleAppsAliases	String		Aliases de domínio (dependendo se o utilizador possui algum perfil ativo em mais do que uma Unidade Orgânica)
ULPrincipalFac	String		Principal Faculdade do utilizador
ULPostalAddr	String	SAP RH, SIGES, GIAF, Fénix, Portal Utilizador	Morada
ULPostalCode	String	SAP RH, SIGES, GIAF, Fénix, Portal Utilizador	Código Postal
ULPUK	String		Senha de acesso provisória para pré-registo da conta
ULRetiredDateADFBA	String		Data que determinado utilizador ficou com perfil inativo na FBA.
ULRetiredDateADULisboa	Date		Data que determinado utilizador ficou com perfil inativo na Reitoria/Serviços Centrais.
ULRetiredDateADCampus	Date		Data que determinado utilizador ficou com todos os perfis inativos em todas as Unidades.
ULResetPassword	String		Atributo usado para guardar temporariamente o token de recuperação de senha.
Surname	String	SAP RH, SIGES, GIAF, Fénix, Portal Utilizador	Apelido
ULSendCredentials	Boolean		Atributo usado pelas equipas de Helpdesk para reenvio de credenciais provisórias de pré-registo da conta.
ULSex	String	SAP RH, SIGES, GIAF, Fénix, Portal Utilizador	Sexo
ULStatus	String		Activo / Inactivo
ULUserID	String	Portal Utilizador	Username escolhido.
ULValidated	Boolean		Atributo que valida o pré-preenchimento de atributos base de forma a que o utilizador esteja em condições de realizar o pré-registo da conta.

Anexo III Atributos da classe ULAuxFac[...]

Nome Atributo	Tipo	Origem	Observações
ULADCtxFac[FAC]	String	AD	Atributo que guarda a localização (DistinguishedName) ao qual o objecto ficou associado.
ULADUserIDFac[FAC]	String	AD	Atributo que guarda o atributo CN que o utilizador tem na respectiva AD.
ULADFac[FAC]	Boolean	AD	Atributo booleano que indica se o objecto está associado com a AD.
ULAnoSemCur[FAC]	String	Fénix, SIGES	Ano Curricular do aluno.
ULArea[FAC]	String	SAP RH, GIAF	Área onde o funcionário desempenha funções.
ULAreaID[FAC]	String	SAP RH	ID da Área onde o funcionário desempenha funções.
ULCourses[FAC]	String	Fénix	Descritivo do nome do curso do aluno.
ULCourseCode[FAC]	String	Fénix, SIGES	Código Curso do aluno.
ULDept[FAC]	String	eDirectory	Atributo preenchido pelo conector Loopback com valores default para cada Unidade Orgânica.
ULDept[FAC]Code	String	eDirectory	Atributo preenchido pelo conector Loopback com valores default para cada Unidade Orgânica.
ULDirecao[FAC]	String	SAP RH, GIAF	Direção de Serviços onde o funcionário desempenha funções.

ULDirecaoID[FAC]	String	SAP RH	ID da Direção de Serviços onde o funcionário desempenha funções.
ULDocenteCode[FAC]	String	SIGES	Código de Docente no Sistema Académico.
ULEmployeeNumber[FAC]	String	SAP RH, GIAF	N.º mecanográfico do funcionário.
ULEstabelecimento[FAC]	String	SAP RH, GIAF	Estabelecimento/Unidade Orgânica do Utilizador
ULExpirationDate[FAC]	Date	Portal Utilizador	Data de expiração da conta (apenas para contas temporárias)
ULExtensao[FAC]	String	SAP RH, GIAF	Extensão telefónica do funcionário.
ULIndCode[FAC]	String	SIGES	Código de indivíduo.
ULInternalEmailAddr[FAC]	String	SAP RH, GIAF, Fénix, SIGES	Endereço de mail interno (referente à Unidade a que pertence) do utilizador.
ULisEmployee[FAC]	Boolean		Indica se o utilizador tem perfil de Funcionário ativo/inativo.
ULisStudent[FAC]	Boolean		Indica se o utilizador tem perfil de Aluno ativo/inativo.
ULisTeacher[FAC]	Boolean		Indica se o utilizador tem perfil de Docente ativo/inativo.
ULisTemporal[FAC]	Boolean		Indica se o utilizador tem perfil de “temporativo” ativo/inativo.
ULLocal[FAC]	String	SAP RH, GIAF	Local físico onde o funcionário desempenha funções.
ULMifare[FAC]	String	SAP RH, GIAF, Fénix	Código Mifare do cartão de identificação.
ULNucleo[FAC]	String	SAP RH, GIAF	Núcleo onde o funcionário desempenha funções.

ULNucleoID[FAC]	String	SAP RH	ID do Núcleo onde o funcionário desempenha funções.
ULRetired[FAC]	Boolean	SAP RH, GIAF, SIGES, Fénix	Atributo que indica que determinado utilizador está ativo/inativo nessa Unidade (é preenchido consoante os atributos booleanos ULisStudent[FAC], ULisTeacher[FAC] e ULisEmployee[FAC] estão todos a true ou falso).
ULStartedDate[FAC]	Date	SAP RH, GIAF, SIGES	Data de inserção do utilizador na BD.
ULStudentCode[FAC]	String	SIGES, Fénix	N.º de Aluno
ULTurmaUnica[FAC]	String	SIGES	Turma principal a que o Aluno pertence.
ULValidated[FAC]	Boolean		Conta validada (pelo sistema) que indica que os atributos default necessários para que o sistema possa criar a conta no repositório Pending e envie as credenciais de pré-registo ao utilizador.

Anexo IV Questionário

Estudo dos utilizadores do Portal do Utilizador da Universidade de Lisboa

A implementação do Portal de Utilizador, insere-se no âmbito de um projeto de gestão de Identidade da ULisboa, componente essencial de criação, recuperação, definição e gestão de acessos dos utilizadores da ULisboa.

A atual interface web de registo, criação e alteração da senha de acesso dos utilizadores da ULisboa, será substituída por um Portal de Utilizador - PU, com novas funcionalidades solicitadas pelos utilizadores, pelas Escolas, decorrente da evolução do projeto de Gestão de Identidade da ULisboa, em curso.

Este questionário servirá de base para o estudo dos utilizadores que irão interagir com o Portal do Utilizador - PU. O seu preenchimento não leva mais do que 10 minutos. A sua participação é essencial para que o portal seja implementado de forma a oferecer funcionalidades que respondam às necessidades dos utilizadores.

A informação recolhida neste questionário, bem como os seus dados pessoais, serão tratados com confidencialidade.

Posteriormente ao seu tratamento, a informação confidencial será destruída.

Para informação adicional, contacte rrito@reitoria.ulisboa.pt

Perfil de Utilizador

1. Idade *

Marcar apenas uma oval.

- <18 anos
- 18 a 20 anos
- 21 a 30 anos
- 31 a 40 anos
- 41 a 50 anos
- 51 a 65 anos
- > 65 anos

2. Sexo *

Marcar apenas uma oval.

- Feminino
- Masculino
- N/S

3. Tipo de Perfil de utilizador *

Marcar tudo o que for aplicável.

- Aluno
- Alumni (ex-aluno)
- Funcionário docente
- Funcionário não-docente
- Antigo funcionário
- Outra: _____

4. Escola/Instituição *

Indicar as Escolas a que pertence/pertenceu

Marcar tudo o que for aplicável.

- Faculdade de Arquitetura (FA)
- Faculdade de Belas-Artes (FBA)
- Faculdade de Ciências (FC)
- Faculdade de Direito (FD)
- Faculdade de Farmácia (FF)
- Faculdade de Letras (FL)
- Faculdade de Medicina (FM)
- Faculdade de Medicina Dentária (FMD)
- Faculdade de Medicina Veterinária (FMV)
- Faculdade de Motricidade Humana (FMH)
- Faculdade de Psicologia (FP)
- Instituto de Ciências Sociais (ICS)
- Instituto de Educação (IE)
- Instituto de Geografia e Ordenamento do Território (IGOT)
- Instituto Superior de Agronomia (ISA)
- Instituto Superior de Ciências Sociais e Políticas (ISCSP)
- Instituto Superior de Economia e Gestão (ISEG)
- Instituto Superior Técnico (IST)
- Reitoria da Universidade de Lisboa (RUL)
- Serviços de Ação Social (SAS)

Conhecimentos de TI do Utilizador

5. Quanto tempo em média por dia costuma utilizar o computador?

Marcar apenas uma oval.

- < 1 hora.
- 1 hora a 2 horas.

- 2 horas a 4 horas.
 - + 4 horas.
6. Da seguinte lista, que contas de e-mail costuma utilizar?
 Marcar tudo o que for aplicável.
- Campus
 - Gmail
 - Office365
 - Outlook
 - Sapo
 - Yahoo
 - Outra:
7. Quais das seguintes ferramentas de TI utiliza regularmente? *
 Marcar tudo o que for aplicável.
- Processamento de texto (ex: Microsoft Word, Google docs)
 - Processamento de dados (ex: Microsoft Excel, Google spreadhseet)
 - Apresentação (ex: Microsoft Power Point, Google presentations)
 - Browser de internet (ex: Chrome, Internet Explorer, Firefox)
 - Edição e manipulação de imagem (ex: Photoshop, Paint)
 - Outra:
8. Tem conhecimento da Conta Campus da ULisboa e dos seus serviços? *
 Marcar apenas uma oval.
- Não (Passe para a pergunta 12).
 - Sim

Serviços Campus

9. Costuma utilizar a conta Campus? *
 Marcar apenas uma oval.
- Não
 - Sim, em várias situações
 - Sim, mas apenas em contexto académico
10. Costuma regularmente alterar a sua senha de acesso em utilizadores.campus.ulisboa.pt? *
 Marcar apenas uma oval.
- Sim
 - Não
 - Não conhecia esta funcionalidade

11. Das seguintes funcionalidades, quais gostaria de ter disponíveis no novo Portal do Utilizador?

Marcar tudo o que for aplicável.

- Ativação de conta com registo de utilizador;
- Alteração de senha de acesso;
- Recuperação da senha de acesso;
- Criação de utilizadores temporários;
- Alteração do e-mail externo para recuperação da senha de acesso;
- Escolha do domínio de e-mail principal
- Ativação de Serviços Google e Office365 (para utilizadores – alunos, docentes e não docentes - das Escolas da ULisboa que não utilizam a conta Campus como conta institucional)
- Acesso a dados pessoais e histórico
- Outra: _____

Hábitos do Utilizador

12. Qual o método de formação preferido para aprender a utilizar aplicações informáticas? *

Marcar tudo o que for aplicável.

- Ler tutoriais
- Ver videos
- Utilizar menu de ajuda da aplicação
- Perguntar a quem já utilizou
- Experimentação/erro
- Outra: _____

13. Onde idealiza utilizar o PU - Portal de Utilizador:

Marcar tudo o que for aplicável.

- Escritório
- Casa
- Sala de aula
- N/S
- Outra:

14. Em que dispositivo idealiza utilizar o PU - Portal de Utilizador?

Marcar tudo o que for aplicável.

- Desktop
- Portátil
- Tablet
- Smartphone

15. Tem por hábito consultar um colega para preencher um formulário online? *
Marcar apenas uma oval.

- Sim
- Não
- Apenas em caso de dúvida

16. Quão frequente é: *
Marcar apenas uma oval por linha.

- Alterar a senha de acesso às aplicações (por ex: e-mail)
- Apenas quando sou forçado pelo sistema a alterar
- Uma vez por ano ou menos
- Entre uma a duas vezes por ano
- Mais do que duas vezes por ano
- N/S

Obrigado

17. Muito obrigado por participar neste questionário.
Caso pretenda receber os resultados deste estudo, deixe o seu e-mail na caixa
abaixo.

Caso pretenda deixar alguma sugestão, use o espaço abaixo para o fazer.

Anexo V Regras das senhas de acesso

Para que a senha de acesso seja válida, tem de obedecer a regras pré-estabelecidas.

A senha de acesso tem de:

- Conter pelo menos 6 caracteres, e no máximo 12;
- Conter pelo menos um carácter de 3 das 4 seguintes categorias:
- Conter pelo menos um carácter maiúsculo: (A até Z)
- Conter pelo menos um carácter minúsculo: (a até z)
- Conter pelo menos um carácter numérico: (0 até 9)
- Conter pelo menos um carácter especial: (~ ! @ # \$ % ^ & * _ - + = ` | () { } } }
: ; " ' < > , . ? /)
- Não poderá fazer referência ao nome do utilizador escolhido nem ao seu nome;
- Não poderá ser igual a nenhuma das 3 últimas senhas de acesso escolhidas (no caso de renovação da conta).

Anexo VI Descrição dos casos de uso

- Caso de Uso: **LOGIN**

Atores: USR, USR-CR, SA-USR

Descrição:

- 1.1- O ator introduz o seu nome de utilizador
- 1.2- O ator introduz a senha de acesso
- 1.3- O ator carrega no botão “ENTRAR”.

- Caso de Uso: **LOGOUT**

Atores: USR, USR-CR, SA-USR

Pré-condição: O ator tem de estar autenticado.

Descrição:

- 1.1- O ator carrega no botão “Logout”.

- Caso de Uso: **ALTERAR SENHA DE ACESSO**

Atores: USR, USR-CR, SA-USR

Pré-condições:

- O ator tem de estar autenticado.
- A senha de acesso tem de obedecer às regras indicadas em Anexo V

Pós-condições:

- A senha é alterada com sucesso

Descrição:

- 1.6- O ator escolhe “ALTERAR SENHA”
- 1.7- O ator insere a sua senha atual
- 1.8- O ator insere a nova senha
- 1.9- O ator repete a nova senha
- 1.10- O ator confirma pressionando “ALTERAR SENHA”.

- Caso de Uso: **ACEITAR TERMOS DE ACEITAÇÃO**

Atores: USR, USR-CR, SA-USR

Pré-condições:

- O ator autenticou-se com sucesso.
- O ator nunca aceitou os termos de aceitação de políticas de acesso, ou os mesmos foram alterados.

Pós-condições:

- É alterado o atributo de aceitação de termos de utilização para aceite.
- É alterada a data que foi realizada a aceitação dos termos de utilização.

Descrição:

- 1.1- O ator carrega em “RECUPERAR ACESSO”
- 1.2- O ator indica o nome de utilizador ou o e-mail externo

▪ Caso de Uso: **RECUPERAR SENHA DE ACESSO**

Atores: USR, USR-CR, SA-USR

Pré-Condições: O ator nunca efetuou o registo de Utilizador escolhendo o nome de utilizar e a senha de acesso.

Pós-condições: É enviado um e-mail para a sua conta de e-mail externo com um *token* de acesso que redireciona para a página de “RESTABELECER SENHA DE ACESSO”

Descrição:

- 1.1- O ator carrega em “RECUPERAR SENHA DE ACESSO”
- 1.2- O ator indica o nome de utilizador ou o e-mail externo

▪ Caso de Uso: **RESTABELECER SENHA DE ACESSO**

Atores: USR, USR-CR, SA-USR

Pré-condições:

- O ator já efetuou o registo de Utilizador escolhendo o nome de utilizar e a senha de acesso.
- O utilizador efetuou um “RECUPERAR SENHA DE ACESSO”
- Foi enviado um e-mail para a conta de e-mail externo do Utilizador um um *token* de acesso que redireciona para a página de “Restabelecer Senha de Acesso”

Descrição:

- 1.1- O ator entre na sua conta de e-mail externo e seleciona o link de enviado para efetuar um “RECUPERAR SENHA DE ACESSO”
- 1.2- O ator indica o nome de utilizador ou o e-mail externo
- 1.3- O ator é redirecionado para o Portal de utilizador, concretamente para a página de “Restabelecer Senha de Acesso” e insere a nova senha.

- 1.4- O ator repete a nova senha
- 1.5- O ator confirma pressionando “RECUPERAR ACESSO”.

- **Caso de Uso: ALTERAÇÃO DO E-MAIL EXTERNO**

Atores: USR, USR-CR, SA-USR

Pré-condições:

- O ator já efetuou o registo de Utilizador escolhendo o nome de utilizar e a senha de acesso.

- O ator autenticou-se com sucesso.

- O ator aceitou os termos de aceitação de políticas de acesso.

Pós-condições: O atributo com o valor do e-mail externo é alterado.

Descrição:

- 1.1- O ator selecionou “DADOS PESSOAIS”

- 1.2- O ator seleciona “ALTERAR E-MAIL EXTERNO”

- 1.3- O ator indica a conta de e-mail externa para recuperação da senha de acesso ou do login.

- **Caso de Uso: ATIVAR CONTA**

Atores: USR

Pré-condições:

- O ator nunca efetuou o registo como Utilizador.

- Foram facultadas ao Utilizador credencias de acesso temporárias

Pós-condições:

- O ator tem um nome de Utilizador e uma senha de acesso para aceder aos serviços.

- O ator passou para o repositório *Users*

Descrição:

- 1.1- O ator insere os dados com as credências provisórias

- 1.2- O ator escolhe um nome de utilizador

- 1.3- O ator indica a senha de acesso

- 1.4- O ator confirma a senha de acesso

- 1.5- O ator seleciona “FINALIZAR REGISTO”

- **Caso de Uso: ACESSO A DADOS PESSOAIS**

Atores: USR, USR-CR, SA-USR

Pré-condições:

- O ator já efetuou o registo de Utilizador escolhendo o nome de utilizador e a senha de acesso.

- O ator tem de estar autenticado

Descrição:

1.1- O ator escolhe “DADOS PESSOAIS”

▪ Caso de Uso: **ATIVAR CONTA GOOGLE FOR EDUCATION E OFFICE365**

Atores: USR

Pré-condições:

- O ator nunca efetuou o registo de utilizador

- O ator tem o atributo ULisOfExternalFac = false

Pós-condições:

- São enviadas para o e-mail externo as credenciais provisórias para o ator

Descrição:

1.1- O ator insere o número de documento de identificação

1.2- O ator seleciona “ATIVAR CONTA EDUCATION”

▪ Caso de Uso: **ALTERAÇÃO DO DOMÍNIO PRINCIPAL DE E-MAIL**

Atores: USR, USR-CR, SA-USR

Pré-condições:

- O ator já efetuou o registo de Utilizador escolhendo o nome de utilizar e a senha de acesso.

- O ator autenticou-se com sucesso.

- O ator aceitou os termos de aceitação de políticas de acesso.

Pós-condições: O atributo com o domínio é alterado.

Descrição:

1.1- O ator selecionou “DADOS PESSOAIS”

1.2- O ator seleciona “ALTERAR DOMÍNIO DE E-EMAIL”, e são apresentados todos os domínios que o ator por escolher.

1.3- O ator escolhe de uma lista de valores o domínio que pretende.

1.4- O ator seleciona a opção “ALTERAR”

▪ Caso de Uso: **CRIAÇÃO DE UTILIZADORES TEMPORÁRIOS**

Atores: USR-CR

Pré-condições:

- O ator já efetuou o registo de Utilizador escolhendo o nome de utilizar e a senha de acesso.
- O ator autenticou-se com sucesso.
- O ator aceitou os termos de aceitação de políticas de acesso.
- O ator tem privilégios para criação de utilizadores temporários numa determinada Escola/Instituição.

Pós-condições:

- É criado um utilizador temporário.
- São enviadas as credências de acesso temporárias para o e-mail indicado.

Descrição:

- 1.1- O ator selecionou “CRIAR UTILIZADOR”
- 1.2- O ator insere os dados no formulário
- 1.3- O ator seleciona a opção “CRIAR UTILIZADOR”

▪ Caso de Uso: **PARAMETRIZAR SISTEMA**

Atores: SA-USR

Pré-condições:

- O ator já efetuou o registo de Utilizador escolhendo o nome de utilizar e a senha de acesso.
- O ator autenticou-se com sucesso.
- O ator aceitou os termos de aceitação de políticas de acesso.
- O ator tem privilégios de Super-Adminstrador.

Pós-condições:

- São registadas as parametrizações.

Descrição:

- 1.1- O ator selecionou “SISTEMA”
- 1.2- O ator seleciona “PARAMETRIZAÇÕES”
- 1.3- O ator efetua as parametrizações no Portal de Utilizador (PU).
- 1.4- O ator seleciona a opção “Atualizar”

▪ Caso de Uso: **VER LOG**

Atores: SA-USR

Pré-condições:

- O ator já efetuou o registo de Utilizador escolhendo o nome de utilizar e a senha de acesso.
- O ator autenticou-se com sucesso.
- O ator aceitou os termos de aceitação de políticas de acesso.

- O ator tem privilégios de Super-Administrador.

Pós-condições:

- São registadas as parametrizações.

Descrição:

1.5- O ator selecionou “SISTEMA”

1.6- O ator seleciona “LOG DE SISTEMA”

▪ Caso de Uso: **CONSULTAR ESTATÍSTICAS**

Atores: SA-USR

Pré-condições:

- O ator já efetuou o registo de Utilizador escolhendo o nome de utilizar e a senha de acesso.

- O ator autenticou-se com sucesso.

- O ator aceitou os termos de aceitação de políticas de acesso.

- O ator tem privilégios de Super-Administrador.

Pós-condições:

- São registadas as parametrizações.

Descrição:

1.1- O ator selecionou “SISTEMA”

1.2- O ator seleciona “DADOS”

Anexo VII Protótipos de baixa fidelidade

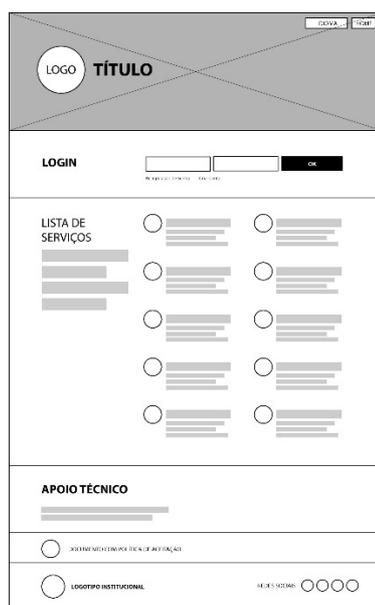


Imagem 40 – Protótipo de interface da página Principal do PU

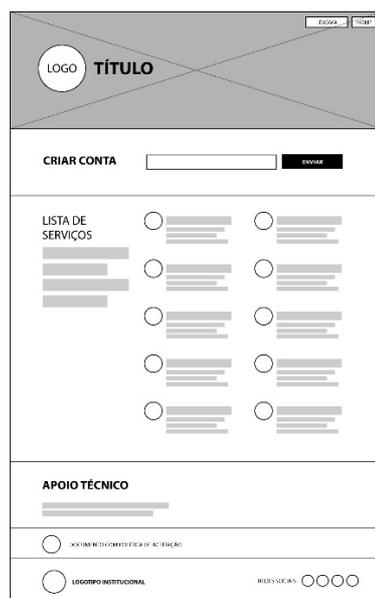


Imagem 41 – Protótipo de interface da página Ativar serviços Google for Education e Office 365



Imagem 42 – Protótipo de interface da página inicial v2

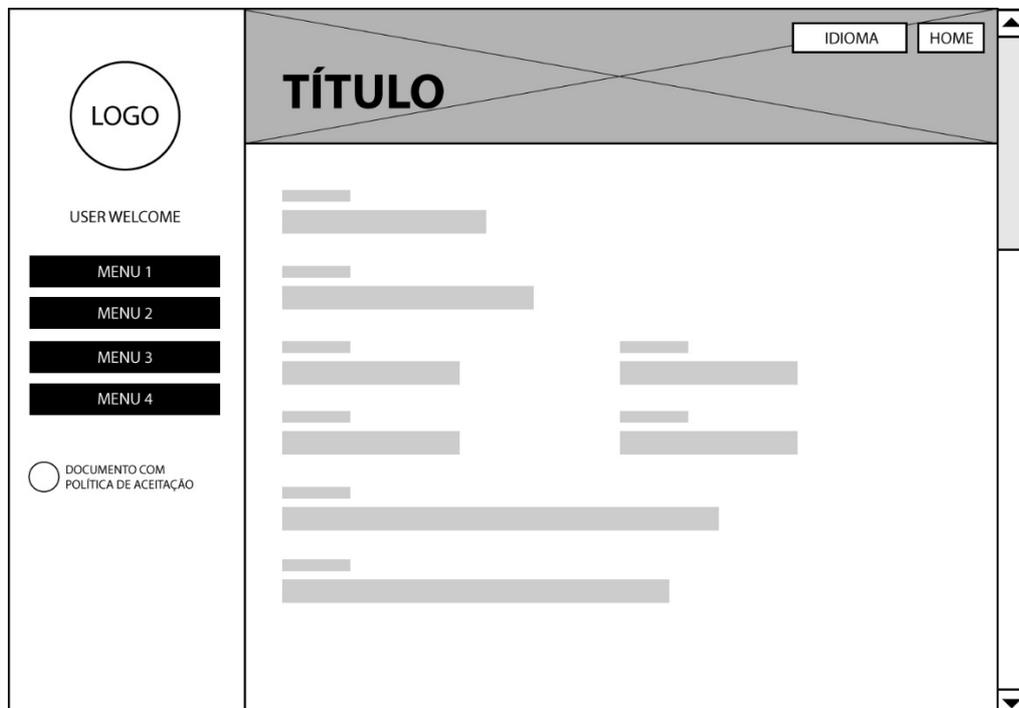


Imagem 43 – Protótipo de interface da página de utilizador com registo feito e opções de navegação



Imagem 44 – Protótipo de interface da página de criação de utilizadores

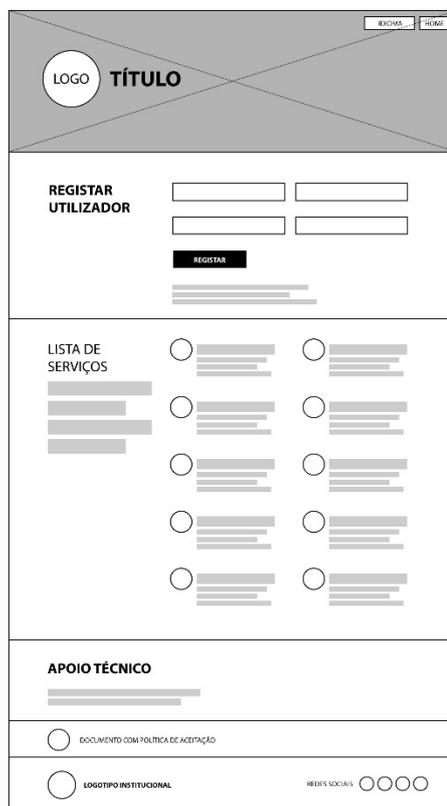


Imagem 45 – Protótipo de interface da página Registar utilizador

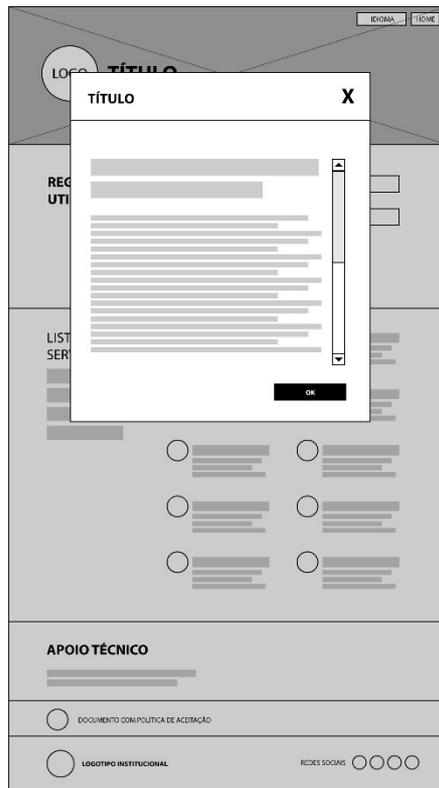


Imagem 46 – Protótipo de interface da página Aceitar Termos de aceitação