

UNIVERSIDADE DE LISBOA

Faculdade de Ciências

Departamento de Informática



SISTEMA DE AJUDA CONTEXTUALIZADA
ONLINE

Steve Michael Pereira Fernandes

MESTRADO EM ENGENHARIA INFORMÁTICA

Sistemas de informação

2011

UNIVERSIDADE DE LISBOA

Faculdade de Ciências

Departamento de Informática



SISTEMA DE AJUDA CONTEXTUALIZADA
ONLINE

Steve Michael Pereira Fernandes
ESTÁGIO

Trabalho orientado pelo Prof. Doutor Pedro Alexandre de Mourão Antunes

e co-orientado por Eng. José Pedro Pereira Cardoso

MESTRADO EM ENGENHARIA INFORMÁTICA
Sistemas de Informação

2011

Agradecimentos

À minha família e amigos por todo o apoio incondicional prestado durante a minha jornada académica. Pela preocupação, cooperação e compreensão que sempre demonstraram, factores que fizeram de mim a pessoa que sou hoje. Sem eles nada disto seria possível.

À Faculdade de Ciências da Universidade de Lisboa e colegas, pelo meu desenvolvimento a nível técnico e pessoal, permitindo-me ganhar asas para encarar sem medo e com confiança a difícil selva profissional dos nossos dias.

À Indra, pelo acolhimento, camaradagem e cuidado demonstrado pela minha formação e bem-estar, proporcionando-me uma adaptação gradual ao mundo do trabalho. Pelo misto de valores dos quais não irei abdicar daqui em diante, o meu muito obrigado.

Gostaria de expressar particularmente os meus sinceros agradecimentos às pessoas que mais directamente se envolveram no desenvolvimento deste estágio:

- Ao Professor Pedro Alexandre de Mourão Antunes pelo acompanhamento académico e inspiração fornecida;
- À Rute Sofia Patrício Arez, ao José Pedro Pereira Cardoso e Frederico Emanuel Moreira Santos pelo suporte (técnico e organizacional) e companheirismo prestado, sempre prontos em me auxiliar e encaminhar na direcção certa;
- Ao João Carlos Seixas Almeida Felisberto e Rui Emanuel Brito De Almeida pelo espírito de entajuda e excelente ambiente de trabalho que me proporcionaram.

Para ti, Hélio Jonilson Van-Dúnem Filipe...

Resumo

O presente relatório descreve o trabalho realizado ao longo de um estágio de nove meses no âmbito da cadeira Projecto em Engenharia Informática.

É bastante usual o utilizador final de uma dada aplicação não possuir um conhecimento prévio das suas potencialidades/funcionalidades, o que por sua vez faz com que seja necessário fornecer um suporte aplicacional que o vise formar na utilização da aplicação.

O projecto foi realizado na plataforma *webMethods* e surgiu justamente para colmatar o tempo gasto no suporte aplicacional e formação dos utilizadores, tendo sido a sua base o melhoramento de um sistema de ajuda contextualizada online para utilizadores de uma aplicação na Web que existia actualmente no Cliente.

Entre os vários aspectos a melhorar estava o facto de a ajuda fornecida aos utilizadores não poder ser editável, o que tornava o sistema bastante estático e desactualizado mal se procedesse a uma alteração da aplicação.

Após as etapas de Análise, Desenho, e Construção, ficou-se com um sistema onde a ajuda é editada online. Cada conteúdo de ajuda ficou conhecido por *Tópico de ajuda*, e por sua vez construiu-se uma árvore de *Tópicos de Ajuda* que representa a navegação possível dentro da aplicação Web. Assim, foi possível realizar adicionalmente um sistema de pesquisa de ajuda em que o resultado é o caminho desde a raiz da aplicação até ao nó onde se encontra o tópico de ajuda com o critério que o utilizador pesquisou.

O sistema abrangeu 3 tipos de utilizadores (Administrador, Técnico e Utilizador), tendo cada um deles funções específicas e acesso a interfaces diferentes. O perfil de Administrador e Técnico pressupõem acesso a interfaces de *BackOffice*, onde se administram os vários projectos que têm um sistema de *Help Online*, sendo também possível fazer uma gestão dos conteúdos de ajuda para cada sistema de ajuda.

O Utilizador por sua vez tem acesso a uma interface de *FrontOffice* onde é fornecida a ajuda sobre a aplicação web e também um sistema de pesquisa.

Actualmente o sistema encontra-se a funcionar no cliente, sendo inclusive utilizado por outras empresas a desenvolverem no local.

Palavras-chave: Ajuda Online, Edição de Conteúdos, Pesquisa de Ajuda, SOA, *webMethods*;

Abstract

This report describes the work done over a nine-month internship for Computer Engineering Project Discipline.

It is quite usual that an end-user of a given application does not have a prior knowledge of its capabilities/features, so it is necessary to provide an application support to train him in the use of the application.

The project was made at the *webMethods* platform with the purpose of filling the time spent on the application support and user training. This was based on an improvement of an online help system for web application users that currently existed in the Client.

Among the different points that needed an improvement was the fact that the Help content provided to the users could not be editable, which made the system easily too static and outdated.

After performing the Analysis, Design and Construction steps, it was reached a system in which help content was editable online. Each one of the contents was named as a Help Topic, and it was part of a tree Help Topic Tree representing the possible navigation within the web application. Thus, it was possible to perform an additional search system where the result was the pathway from the web application root to the node with a help content that match the search key.

The system has three types of users (Administrator, Technical and User), each one having specific functions and having access to different interfaces. The Administrator and Technical profile implies access to the Back office interfaces, where one can manage multiple projects (web Applications) that would have an Help Online system, and it is also possible to make a content management to each help system. The user has access to an Front Office interface that provides help to each page, and also a web search system.

Currently the system is running on the client, and it is also being used by other companies working on the client.

Keywords: Help Online, Content Management, Help Search System, SOA, *Webmethods*

Conteúdo

Capítulo 1	Introdução.....	13
1.1	Contexto	13
1.2	Integração na empresa	15
1.3	Integração no projecto	16
1.4	Objectivos.....	17
1.5	Organização do Documento	17
Capítulo 2	Contexto do projecto	19
2.1	Visão.....	19
2.2	Missão.....	19
2.3	Objectivos Operacionais.....	20
2.4	Metodologia.....	21
2.4.1	Estudo de viabilidade do sistema (EVS)	21
2.4.2	Análise do sistema de informação (ASI).....	22
2.4.3	Desenho do sistema de informação (DSI).....	23
2.4.4	Construção do sistema de informação (CSI).....	24
2.4.5	Implementação e aceitação do sistema (IAS)	25
2.5	Contexto Tecnológico	27
2.5.1	A Plataforma <i>Webmethods</i>	27
2.5.2	Arquitectura lógica.....	33
2.5.3	Arquitectura Física	35
2.5.4	Ferramentas auxiliares.....	36
Capítulo 3	Trabalho realizado.....	37
3.1	Gestão de actividades	37
3.1.1	Calendarização	38
3.2	Estudo de viabilidade do sistema (EVS)	40
3.3	Análise do sistema de informação (ASI).....	41

3.3.1	Requisitos Funcionais	41
3.3.2	Diagramas Caso de Uso	42
3.3.3	Requisitos Não Funcionais.....	44
3.3.4	Especificação de Ecrã.....	45
3.4	Desenho do sistema de informação (DSI)	47
3.4.1	Modelo Relacional [6].....	47
3.4.2	Exemplo de Descrição de Serviço.....	47
3.5	Construção do sistema de informação (CSI)	49
3.5.1	Camada de Apresentação	50
3.5.2	Camada de Lógica de Negócio.....	53
3.5.3	Camada de Dados.....	57
3.5.4	Testes.....	59
3.6	Implementação e aceitação do sistema (IAS).....	61
Capítulo 4	Conclusão	73
Capítulo 5	Bibliografia.....	75
Capítulo 6	Anexos.....	77

Lista de Figuras

Fig. 1: Organigrama Indra Sistemas Portugal.....	14
Fig. 2: Organização do sector de Soluções Tecnológicas.....	16
Fig. 3: Metodologia de Desenvolvimento.....	21
Fig. 4: Metodologias - Análise do Sistema de informação.....	22
Fig. 5: Metodologias - Desenho do Sistema de informação.....	23
Fig. 6: Metodologias - Construção do Sistema de informação.....	24
Fig. 7: Metodologias – Implementação e aceitação do Sistema de informação.....	26
Fig. 8: Modelo em três camadas (mais Clientes Web) do <i>Webmethods</i>	28
Fig. 9: Arquitectura do <i>Webmethods</i>	29
Fig. 10: Interface de desenvolvimento do <i>Webmethods Designer</i>	30
Fig. 11: Interface Web do <i>MyWebmethods Server</i>	31
Fig. 12: Interface de desenvolvimento do <i>Webmethods Developer</i>	32
Fig. 13: Interface Web do <i>Integration Server</i>	33
Fig. 14: Arquitectura Lógica.....	34
Fig. 15: Exemplo de Arquitectura Física.....	35
Fig. 16: Caso de Uso – Interface de administração.....	42
Fig. 17: Caso de Uso – Interface de gestão.....	42
Fig. 18: Caso de Uso – Interface de utilizador.....	43
Fig. 19: Ecrãs do sistema <i>Help Online</i>	45
Fig. 20: Modelo Relacional de Dados.....	47
Fig. 21: Exemplo de construção de interface no <i>Webmethods Designer</i>	51
Fig. 22: Exemplo do Serviço ‘inserirTemplDespacho’ no <i>Developer</i>	55
Fig. 23: Exemplo de um <i>Adapter Service</i> para Criar um Projecto.....	56
Fig. 24: Diagrama Requisitos vs Produto.....	60
Fig. 25: Help Online Demo – Interface de Administração.....	61
Fig. 26: Help Online Demo – <i>Popup</i> de Adicionar Novo Projecto.....	62
Fig. 27: Help Online Demo – <i>Popup</i> de Criação de Tópicos de ajuda (tab 1).....	62

Fig. 28: Help Online Demo – <i>Popup</i> de Criação de Tópicos de ajuda (tab 2)	63
Fig. 29: Help Online Demo – <i>Popup</i> de Criação de Tópicos de ajuda (tab 3)	63
Fig. 30: Help Online Demo – <i>Popup</i> de Edição de Tópicos de ajuda (tab 1).....	64
Fig. 31: Help Online Demo – <i>Popup</i> de Edição de Tópicos de ajuda (tab 2).....	64
Fig. 32: Help Online Demo – <i>Popup</i> de Edição de Tópicos de ajuda (tab 3).....	65
Fig. 33: Help Online Demo – Relatório de resumo do projecto	65
Fig. 34: Help Online Demo – Interface de Gestão.....	66
Fig. 35: Help Online Demo – <i>Popup</i> de carregamento de Imagens	67
Fig. 36: Help Online Demo – <i>Popup</i> de Pesquisa de Imagem.....	68
Fig. 37: Help Online Demo – Ecrã Editar Tópico de Ajuda.....	69
Fig. 38: Help Online Demo – <i>Popup</i> Escolha Manual de Imagem.....	69
Fig. 39: Help Online Demo – <i>Popup</i> Escolha De Imagem por Pesquisa.....	69
Fig. 40: Help Online Demo – Ecrã de Consulta de Tópicos de Ajuda (1).....	71
Fig. 41: Help Online Demo – Ecrã de Consulta de Tópicos de Ajuda (2).....	71
Fig. 41: Help Online Demo – Ecrã de Consulta de Tópicos de Ajuda (3).....	71
Fig. 42: Help Online Demo – Ecrã de Consulta de Tópicos de Ajuda (4).....	71

Lista de Tabelas

Tabela 1: Perfis dos colaboradores Indra.....	15
Tabela 2: Metodologias - Tarefas de Análise do Sistema de Informação	23
Tabela 3: Metodologias - Tarefas de Desenho do Sistema de Informação.....	24
Tabela 4: Metodologias - Tarefas de Construção do Sistema de Informação	25
Tabela 5: Metodologias - Tarefas de IAS do Sistema de Informação	26
Tabela 6: Exemplo de Requisitos Funcionais na estrutura do documento	41
Tabela 7: Exemplo de descrição de Caso de Uso	43
Tabela 8: Exemplo de descrição de Caso de Uso	44
Tabela 9: Exemplos de Requisitos Não Funcionais na estrutura do documento	44
Tabela 10: Exemplo de especificação de um ecrã	46
Tabela 11: Exemplo de Descrição do serviço <i>criarProjecto</i>	48
Tabela 12: Exemplo de Descrição do serviço <i>criarTopicoAjuda</i>	48
Tabela 13: Exemplo de Código escrito em Java	52
Tabela 14: Stored Procedure Specification (sps)	57
Tabela 15: Stored Procedure Body (spb)	58

Lista de Acrónimos

Acrónimo	Extenso
<i>Ajax</i>	<i>Asynchronous Javascript and XML</i>
<i>BD</i>	<i>Base de Dados</i>
<i>BPM</i>	<i>Business Process Management</i>
<i>CAF</i>	<i>Composite ApplicationFramework</i>
<i>GUI</i>	<i>Graphical User Interface</i>
<i>HTTP</i>	<i>Hypertext Transfer Protocol</i>
<i>I+D</i>	<i>Investigação + Desenvolvimento</i>
<i>IDE</i>	<i>Integrated Development Environment</i>
<i>IS</i>	<i>Integration Server</i>
<i>JDBC</i>	<i>Java Database Connectivity</i>
<i>JSF</i>	<i>Java Server Faces</i>
<i>MWS</i>	<i>MyWebMethods</i>
<i>RPC</i>	<i>Remote Procedure Call</i>
<i>SGBD</i>	<i>Sistema de Gestão de Base de Dados</i>
<i>SOA</i>	<i>Service Oriented Architecture</i>
<i>SOAP</i>	<i>Simple Object Access Protocol</i>
<i>SQL</i>	<i>Structured Query Language</i>
<i>SSH</i>	<i>Secure Shell</i>
<i>SVN</i>	<i>SubVersion</i>
<i>XML</i>	<i>eXtensible Markup Language</i>

Capítulo 1

Introdução

1.1 Contexto

O estágio em que me inseri teve início a 6 de Setembro de 2010 e uma duração total de 9 meses, tendo sido desenvolvido o projecto nas próprias instalações do cliente que facultou os serviços disponibilizados pela Indra Sistemas Portugal, S.A.

A orientação na empresa esteve a cargo do Eng. José Pedro Pereira Cardoso, que me acompanhou ao longo do estágio e forneceu o suporte necessário ao desenvolvimento do projecto e integração na tecnologia.

A INDRA Sistemas é uma empresa multinacional de Tecnologias de Informação, número um em Espanha e uma das principais na Europa e na América Latina. A sua capitalização bolsista coloca-a entre as três maiores empresas europeias do seu sector sendo uma das três empresas espanholas que mais investe em I+D. Actualmente a Indra conta com mais de 23.500 colaboradores e tem clientes em mais de 90 países.

Em Portugal a empresa é composta por uma equipa com cerca de 400 profissionais com elevada especialização, procurando ser o principal aliado dos seus clientes no uso intensivo das tecnologias de informação.

A estrutura da empresa divide-se em vários mercados verticais que lhe permitem criar uma oferta diferencial potenciando o desenvolvimento de soluções próprias para cada segmento de mercado, como por exemplo: Administração Pública e Saúde, Transporte e tráfego, Finanças e Seguros, Energia e Industria Telecomunicações e Media, e Defesa.

A Indra presta serviços que vão desde consultoria, desenvolvimento de projectos, integração de sistemas e aplicações até ao outsourcing de sistemas de informação e de processos de negócios.

A oferta de soluções inclui uma ampla gama de sistemas, aplicações e componentes para a captação de dados e informação, o seu tratamento, transmissão e posterior apresentação, basicamente direccionados para o controle e gestão de processos complexos e/ou críticos.

De um modo geral, as soluções da Indra servem o núcleo de operação do negocio dos seus clientes e requerem como capacidade essencial a integração de sistemas, isto é, o desenho, configuração, desenvolvimento e implementação de componentes, aplicações e sistemas de informação completos, incorporando produtos próprios e de terceiros.

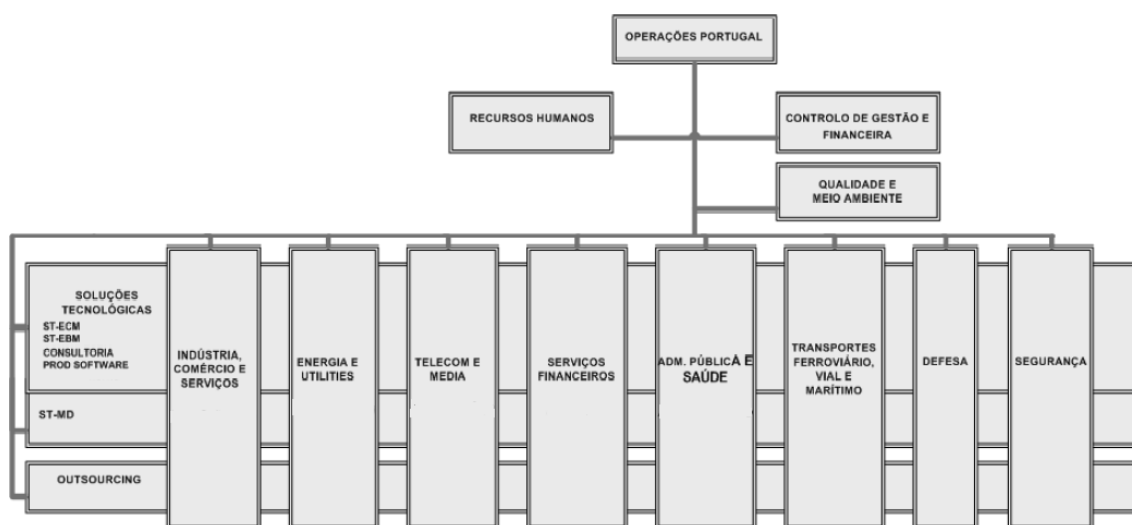


Fig. 1: Organograma Indra Sistemas Portugal

De seguida fica uma descrição mais detalhada sobre cada perfil inerente às equipas Indra:

Perfil	Descrição
Director de Projecto	<ul style="list-style-type: none"> ▪ Boa compreensão das realidades e orientações da organização; ▪ Fortes competências na direcção de projectos;
Gestor de Projecto	<ul style="list-style-type: none"> ▪ Experiência em projectos de construção de arquiteturas de integração e de implementação de soluções; ▪ Conhecimento detalhado das metodologias de projecto seleccionadas pela Indra; ▪ Experiência na construção e seguimento de modelos de avaliação de benefícios; ▪ Experiência na articulação com parceiros numa óptica institucional e de projecto.

Perfil	Descrição
Consultor de Qualidade	<ul style="list-style-type: none"> ▪ Conhecimento detalhado das metodologias de projecto e dos procedimentos de controlo de qualidade da Indra; ▪ Comprovada experiência na verificação de qualidade de projectos nas distintas vertentes – Gestão, Processos, Tecnologia, Pessoas e Benefícios
Coordenador Técnico (<i>Expert</i>)	<ul style="list-style-type: none"> ▪ Experiência na liderança operacional de equipas de implementação; ▪ Comprovada experiência em projectos de integração e no desenho de arquitecturas tecnológicas; ▪ Especialista na implementação de soluções baseadas em <i>webMethods</i>; ▪ Experiência em análise e desenho de processos;
Consultor Sénior Tecnológico	<ul style="list-style-type: none"> ▪ Experiência na liderança operacional de equipas de implementação; ▪ Comprovada experiência em projectos de integração e no desenho de arquitecturas tecnológicas; ▪ Experiência na implementação de soluções baseadas em Java; ▪ Experiência em análise e desenho de processos;
Consultor/Engenheiro de sistemas	<ul style="list-style-type: none"> ▪ Forte conhecimento das ferramentas a utilizar no projecto, privilegiando-se a certificação nos mesmos; ▪ Experiência no desenho e desenvolvimento de interfaces; ▪ Experiência de programação em tecnologias utilizadas no projecto; ▪ Experiência na utilização de <i>webservices</i>; ▪ Experiência no desenho técnico e modelação de processos de <i>workflow</i>.

Tabela 1: Perfis dos colaboradores Indra

1.2 Integração na empresa

Na primeira semana tive oportunidade de participar numa sessão de acolhimento cujo objectivo foi integrar os novos colaboradores, principalmente aqueles que como eu não tinham experiência profissional, possibilitando assim uma adaptação gradual ao longo de uma semana. Esse evento contou com 3 dias de formação proporcionados por uma empresa contratada para o efeito.

Os dois primeiros dias serviram particularmente para conhecer melhor a empresa, a sua estrutura, os projectos em que participa e as soluções de que dispõem. Para tal foi feita uma apresentação individual por parte de cada director de cada mercado.

Nos últimos 3 dias fomos presentes a várias formações que visavam desenvolverem as nossas *soft-skills* para ingressar no mundo profissional. Aprendemos técnicas para trabalhar em equipa, técnicas de comunicação e técnicas para melhorar a nossa imagem.

1.3 Integração no projecto

Com o cargo de Engenheiro de Sistemas Júnior, o meu trabalho inseriu-se quer na vertente técnica, quer na vertente funcional, tendo sido alocado no sector de Soluções Tecnológicas descrito na imagem abaixo.

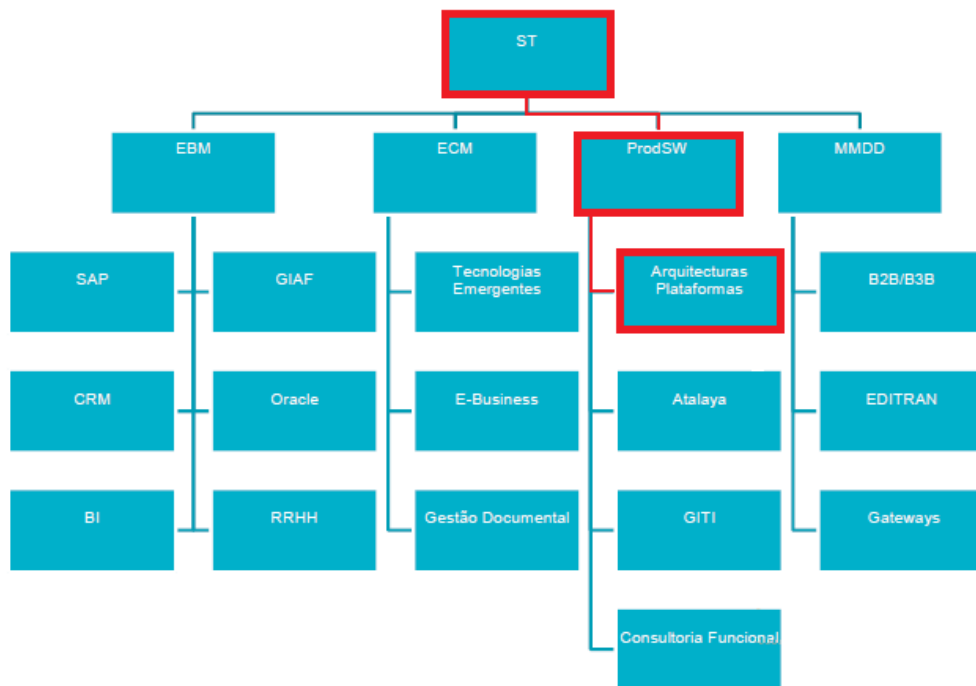


Fig. 2: Organização do sector de Soluções Tecnológicas

Apesar do cargo sugerir apenas um trabalho de perfil técnico, foi acordado com o Gestor de Projecto a minha participação em trabalhos funcionais, tal como o planeamento enviado para a FCUL sugeria.

A minha integração no projecto passou pela leitura dos documentos relativos a cada projecto a ser desenvolvido na altura. Posto isto deu-se a realização de várias tarefas de ambientação à tecnológica, como a realização de pequenos tutoriais, serviços, etc.

1.4 Objectivos

Tratando-se de um Estágio numa empresa que realiza serviços de consultoria, existiu sempre a possibilidade de os objectivos não estarem focados a um único projecto mas sim a uma tecnologia. No meu caso específico tive a possibilidade de me concentrar unicamente num projecto. Sendo assim, estes foram os objectivos identificados no início do estágio:

- Efectuar a análise, desenho e implementação de numa plataforma tecnológica, compreendendo primeiramente os problemas e necessidades que deram origem ao projecto;
- Desenvolvimento de um sistema através da plataforma *webMethods*, nomeadamente componentes da interface gráfica e *webservices* para uso da mesma, recorrendo às ferramentas apresentadas pela plataforma em questão (*webMethods Designer e webMethods Developer*);
- Fazer análise crítica dos métodos de desenvolvimento aplicados, procurando sempre melhorar a utilização da plataforma *webMethods*;
- Executar casos de testes para validação funcional e técnica dos processos implementados;
- Ter contacto directo com o cliente através de reuniões periódicas.

1.5 Organização do Documento

Este documento encontra-se organizado em 4 capítulos que são elucidados de seguida:

- No **Capítulo 1** é feita uma introdução ao documento, deixando claro o seu contexto. É apresentada a empresa onde se realiza o estágio e são identificados os objectivos;
- No **Capítulo 2** é feita uma contextualização sobre as condições que deram origem ao trabalho desenvolvido, descrevendo a tecnologia/arquitectura usada, assim como as metodologias para o seu desenvolvimento;
- No **Capítulo 3** comentou-se o trabalho que foi realizado, quer na vertente mais técnica, quer na vertente de gestão das actividades, mostrando no fim o resultado tangível de todo o estágio;
- O **Capítulo 4** conclui este relatório, dando uma perspectiva final sobre o estágio desde o processo de acolhimento até à sua conclusão;

Capítulo 2

Contexto do projecto

Este projecto teve por base o melhoramento de um sistema que já estava implementado no cliente mas que se encontrava obsoleto e não ia ao encontro dos objectivos pretendidos. Foram propostas várias soluções, sendo elas elaboradas e completadas à medida que se iam realizando reuniões e se percebia a dimensão do problema assim como qual seria a solução final pretendida. De seguida é feita uma contextualização sobre o surgimento do projecto.

2.1 Visão

Muitas das vezes um utilizador final de um sistema não é aquele que impôs os requisitos do projecto e especificou as funcionalidades pretendidas para o mesmo. Sendo assim é perfeitamente compreensível que ele não possua um conhecimento profundo sobre o sistema e que por sua vez não conheça as funcionalidades/potencialidades de que pode usufruir, o que normalmente é colmatado na fase de Suporte Pós-implementação / Manutenção por quem desenvolveu o sistema ou por quem tem algum conhecimento sobre ele. Este passo é sempre tido em conta no planeamento de um projecto, mas é justamente sobre ele que incidiu o estudo, fornecendo um sistema de ajuda contextualizada para reduzir o tempo gasto no Suporte, podendo assim efectuar outras actividades consideradas mais importantes para a empresa.

2.2 Missão

Este projecto surge para fornecer um sistema de ajuda on-line contextualizada que se adapte a todos os projectos feitos em *webMethods*. A ajuda fornecida é contextualizada e pré-definida, tendo um principal foco sobre situações que levam o utilizador a ficar com dúvidas sobre o que está ou o que deve fazer, substituindo assim o

Suporte que muitas das vezes se resumia a uma simples indecisão do utilizador sobre uma funcionalidade de um ecrã.

Embora existisse um antigo sistema de Ajuda Online, esse sistema representava uma solução não genérica e que não se adaptava aos vários projectos a serem desenvolvidos, não indo assim de encontro aos objectivos do Cliente (explicados no ponto seguinte). No sistema antigo eram fornecidos manuais de utilização (previamente concebidos por uma equipa funcional) para cada página de uma aplicação web (por exemplo) através de um link de ‘ajuda’ disponível, despoletando assim o respectivo download do documento para posterior leitura por parte do utilizador. Entre várias limitações identificadas [4] [5], estava o facto de o utilizador ter que sair da aplicação para abrir o documento.

2.3 Objectivos Operacionais

O objectivo principal, que seria fornecer ajuda contextualizada para a aplicação, já estava implícito mas de facto o que o cliente pretendia era algo diferente, dando assim origem à lista de requisitos proveniente das reuniões onde se discutiram os prós e contra do Sistema antigo, fornecendo ao mesmo tempo ideias inovadoras para o que viria a ser o novo sistema de *Help Online*. De seguida são mostrados os principais requisitos que acabam por definir o objectivo do cliente para este projecto:

- Ter um sistema que disponibilize ajuda para cada ecrã da aplicação;
- Ter um sistema em que seja possível editar a ajuda presente em cada ecrã;
- A ajuda tanto pode ser relativa a um ecrã como a um componente desse ecrã (exemplo: botão);
- O sistema de ajuda tem que ser ajustável a todos os projectos desenvolvidos na ferramenta *WebMethods*;
- Ter no *Help Online* uma opção de pesquisa sobre uma ajuda;

Averiguada a grande quantidade e complexidade de ecrãs que um projecto pode possuir, e descritos os principais objectivos para o projecto, podemos afirmar que o *Help Online* terá os seguintes resultados tangíveis:

- Ganhos de tempo por parte da equipa: ao invés de ocupar tempo ajudando utilizadores ou disponibilizando formação, poderão ocupar o tempo com outras tarefas;

- Ganhos de tempo por parte do utilizador: ao invés de solicitar ajuda á equipa de desenvolvimento poderá resolver os problemas por si mesmo;
- Ganhos para a empresa: visto o projecto ser adaptável a todas as aplicações *webMethods*;

2.4 Metodologia

Neste capítulo é descrita a metodologia usada, servindo o esquema seguinte para uma visualização mais explícita das actividades que decorrem em simultâneo nas Metodologia de Desenvolvimento e de Gestão de Projecto. De referir que nem todas as tarefas dentro de cada etapa foram realizadas, ajustando assim a metodologia à realidade do projecto.

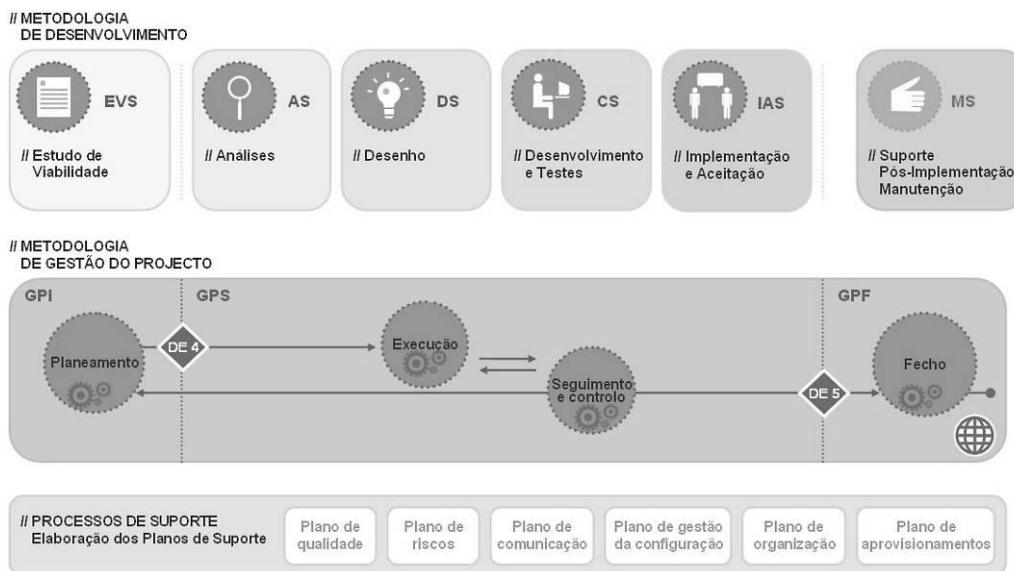


Fig. 3: Metodologia de Desenvolvimento

2.4.1 Estudo de viabilidade do sistema (EVS)

O objectivo desta etapa é analisar um conjunto concreto de necessidades (económicas, técnicas, legais e operativas), com a ideia de propor uma solução a curto prazo, sendo ela responsável pela Avaliação do tempo e recursos necessários.

Os resultados do Estudo de Viabilidade do Sistema constituirão a base para tomar a decisão de seguir em frente ou de abandonar.

Este processo é considerado importante, embora o nível de profundidade dependa de cada caso. Como o sistema de Help Online foi um pedido feito directamente pelo Cliente e visto já existir um sistema base, esta fase foi constituída por reuniões para propor a solução.

2.4.2 Análise do sistema de informação (ASI)

O propósito desta etapa é conseguir a especificação detalhada do Sistema de Informação através de um catálogo de requisitos e de uma série de modelos que cubram as necessidades de informação dos utilizadores para os quais se desenvolverá o sistema de informação.

Neste processo inicia-se também a especificação do Plano de Testes, que se completará no processo Desenho do Sistema de Informação (DSI).

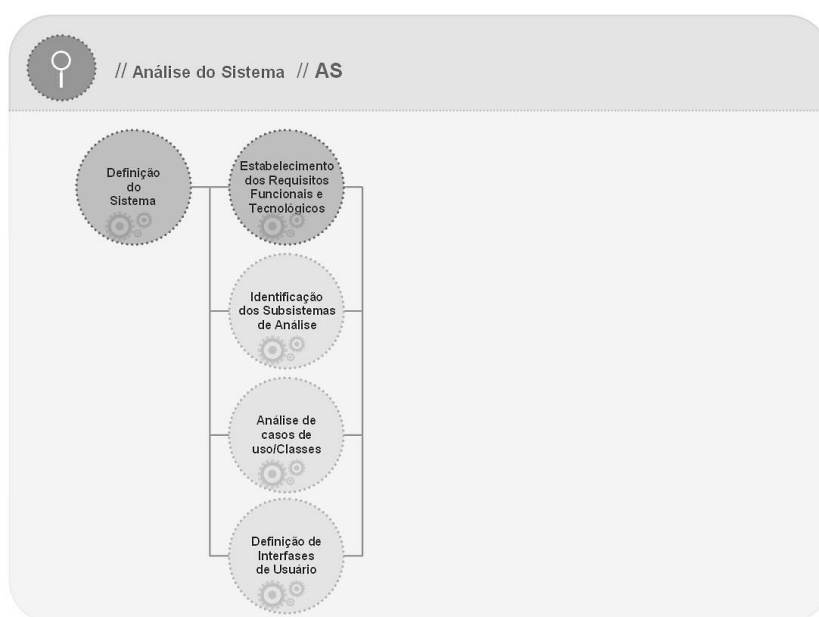


Fig. 4: Metodologias - Análise do Sistema de informação

As principais tarefas a realizar de acordo com o diagrama acima e enquadrados no contexto do projecto poderão ser:

Actividades	Outputs	Tarefa
Definição do sistema	Catálogo de requisitos gerais	Determinar o alcance do sistema e utilizadores finais
Identificação de Requisitos Funcionais e não Funcionais	Catálogo de requisitos (modelo funcional)	Obtenção e análise de Requisitos
Identificação de Subsistemas de Análise		Determinação e integração de subsistemas de análise

Actividades	Outputs	Tarefa
Definição de Interfaces de Utilizador	(modelo funcional)	Especificação de Princípios Gerais do Interface
Análise de Consistência e Especificação de Requisitos	Especificação de Requisitos Software	Especificação e análise dos modelos Elaboração da Especificação de Requisitos Software (ERS)
Especificação do Plano de Testes		Definição do alcance dos Testes
Aprovação do AS		Apresentação e Aprovação do AS

Tabela 2: Metodologias - Tarefas de Análise do Sistema de Informação

2.4.3 Desenho do sistema de informação (DSI)

O objectivo do Desenho do Sistema de Informação (DSI) é obter a definição da arquitectura do sistema e do ambiente tecnológico que lhe irá dar suporte, junto com a especificação detalhada dos componentes do sistema de informação.

O desenho da arquitectura do sistema dependerá em grande medida das características da instalação requerendo a participação dos responsáveis de SI das Organizações para as quais se desenvolve o sistema de informação.

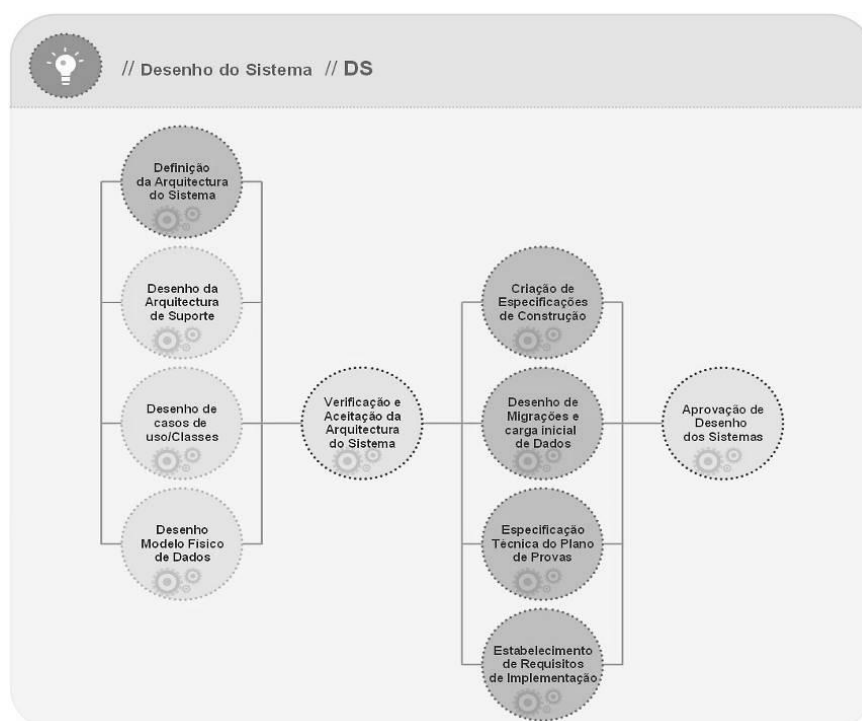


Fig. 5: Metodologias - Desenho do Sistema de informação

Seguidamente são apresentadas as principais tarefas e respectivos Outputs:

Actividades	Outputs	Tarefa
Desenho da Arquitectura	Desenho Detalhado de Subsistemas	Desenho de Subsistemas de Suporte
Desenho Físico de Dados	Modelo Físico de Dados	Desenho do Modelo Físico de Dados (MFD)
Desenho da Migração e Carregamento Inicial de Dados	Plano de Migração e Carregamento Inicial de Dados	Desenho de Procedimentos e componentes de Migração e Carregamento Inicial
Especificação Técnica do Plano de Testes	Doc. de Planificação dos Testes	Especificação do Ambiente de Testes e respectivo plano
Estabelecimento de requisitos de Implementação	Requisitos de Implementação	Especificação de requisitos de implementação
Aprovação do Desenho do sistema		Apresentação e Aprovação do DS

Tabela 3: Metodologias - Tarefas de Desenho do Sistema de Informação

2.4.4 Construção do sistema de informação (CSI)

A construção do Sistema de Informação (CSI) tem como objectivo final a construção e teste dos diferentes componentes do sistema de informação tendo por base o Processo de Desenho do Sistema de Informação (DSI). Prepara-se o ambiente de construção, cria-se o código de cada um dos componentes do sistema de informação e vai-se realizando, à medida que se vai finalizando a construção, os testes unitários de cada um deles e os de integração entre subsistemas.

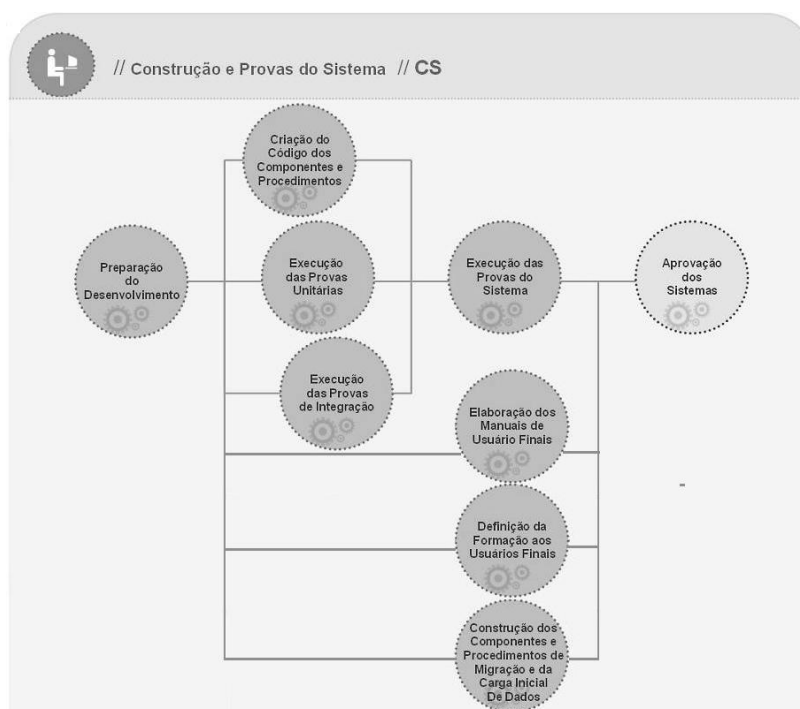


Fig. 6: Metodologias - Construção do Sistema de informação

Seguidamente são apresentadas as principais tarefas e respectivos Outputs:

Actividades	Outputs	Tarefa
Desenvolvimento do Código de componentes e procedimentos	Código Fonte de Componentes	Geração do Código dos componentes e Procedimentos de Operação e Segurança.
Execução de Testes Unitários	Relatórios de Testes unitários	Realização de Testes Unitários Avaliação de Testes Unitários
Execução de Testes de Integração	Relatório de Testes de integração	Realização e Avaliação dos Testes de Integração;
Execução de Testes de Sistema	Avaliação e Resultado dos Testes ao Sistema	Realização e Avaliação dos Testes de Sistema.
Elaboração dos Manuais de Utilizador	Manuais de Utilizador	Elaboração dos Manuais de Utilizador
Construção de Componentes, Construção de Procedimentos de Migração e Carregamento Inicial de Dados	Procedimentos de Operação, Administração do Sistema, Segurança e Controlo de Acesso	Preparação da migração e carregamento inicial de dados; Geração do código dos Componentes e Procedimentos de Migração e Carregamento Inicial de Dados;
Aprovação do sistema		Apresentação e Aprovação do sistema

Tabela 4: Metodologias - Tarefas de Construção do Sistema de Informação

2.4.5 Implementação e aceitação do sistema (IAS)

Este processo tem como objectivo principal a entrega e aceitação do sistema na sua totalidade e a realização das actividades necessárias para a passagem à produção do sistema.

O Sistema irá submeter-se aos Testes de Implementação com a participação do utilizador, cuja responsabilidade, entre outros aspectos, é comprovar o comportamento do sistema sob as condições mais extremas. Também se submeterá aos Testes de Aceitação, cuja execução é da responsabilidade do utilizador final.

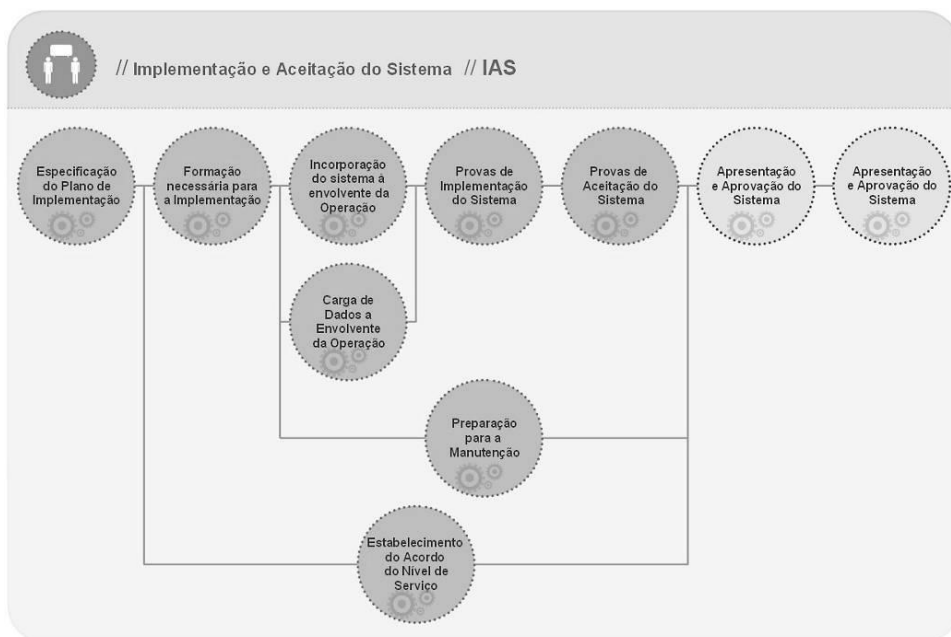


Fig. 7: Metodologias – Implementação e aceitação do Sistema de informação

As principais tarefas a realizar de acordo com o diagrama acima e enquadrados no contexto do projecto e proposta poderão ser:

Actividades	Outputs	Tarefa
Formação necessária para a Implementação	Plano de Formação para Utilizadores Finais e Equipa de Implementação	Preparação da Formação da Equipa de Implementação. Preparação da formação a Utilizadores Finais.
Carregamento de Dados ao Ambiente de Operação		Migração e Carregamento inicial de Dados
Testes de Implementação do sistema	Relatório e resultado dos Testes de Implementação	Preparação, realização e avaliação dos Testes de Implementação.
Testes de Aceitação do sistema	Relatório e resultado dos Testes de Aceitação	Preparação, realização e avaliação dos Testes de Aceitação
Apresentação e Aprovação do sistema	BD e Ficheiros Carregados; Aprovação do Sistema; Emissão de Certificado de Aceitação por parte do Cliente	Convocatória da Apresentação do Sistema Aprovação do Sistema
Passagem a Produção	Sistema em produção	Preparação e activação do Ambiente de Produção

Tabela 5: Metodologias - Tarefas de IAS do Sistema de Informação

2.5 Contexto Tecnológico

A empresa *webMethods* foi fundada em 1996 por Phillip Merrick, sendo adquirida pela *Software AG* em 2007, empresa essa que dispõem actualmente de um conjunto de produtos honrando o nome de origem. A suite *webMethods* caracteriza-se como sendo um software para melhoria de processos, disponibilização de **SOA** [13] e principalmente integração.

Durante os seus primeiros anos a empresa faliu, mas com uma infusão de capital e alguns novos clientes como o *Citi Bank*, *Fedex* e *Dell* viria a ganhar novo fôlego. Um pouco mais tarde formalizou-se uma parceria com a Microsoft para desenvolver a **XML Query Language**.

No ano seguinte desenvolveu outra parceria estratégica, desta vez com a *SAP*, para fornecer capacidades de integração entre as várias tecnologias. Com todas estas parcerias a *webMethods* estabeleceu-se como um fornecedor de *middleware* com soluções de integração de aplicações, integração de parceiros e produtos de **BPM**.

Com o adquirir da *Webmethods*, a *Software AG* fez dessa ferramenta uma das suas principais linhas de produtos.

2.5.1 A Plataforma *Webmethods*

A plataforma *webMethods* pode entender-se como uma tecnologia orientada a serviços que se enquadra na arquitectura *SOA* e no conceito *BPM*. Abaixo são referenciados alguns tópicos que considero interessantes para introduzir esta tecnologia:

- Uso de uma Arquitectura Orientada pelos Serviços (SOA) [13] em que as aplicações desenvolvidas devem usar funcionalidades oferecidas em forma de serviços de um sistema computacional diferente, sendo que estes serviços poderão ser usados por várias aplicações;
- Possibilidade de modelação gráfica de processos (BPM) através da sua decomposição em tarefas humanas e/ou integração de serviços aplicativos. Desta forma o cliente deterá uma plataforma de gestão de processos ágil e flexível, simplificando a manutenção dos processos actuais, bem como a criação de novos processos;

- Disponibilização de *webServices*, acessíveis via Web (**HTTP**) e que fornecem para o exterior funcionalidades ou serviços, de forma a poderem ser usados por um cliente;
- Uso de **XML**;

Os pontos descritos em cima privilegiam maioritariamente a facilidade de manutenção, a possibilidade de expansão e a capacidade de integração de novas funcionalidades, com o intuito de permitir a evolução do sistema ao longo do tempo, de acordo com as necessidades.

A arquitectura pode ser vista segundo três níveis principais [7], sendo eles a Camada de Apresentação, a Camada de Lógica de Negócio e a Camada de Dados, conforme alude a figura abaixo:

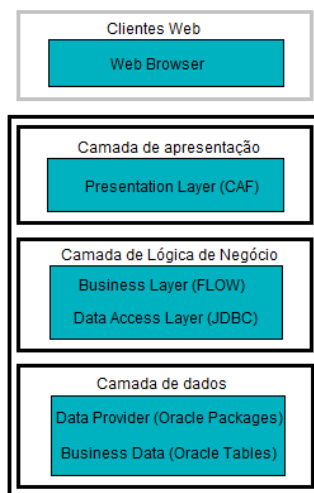


Fig. 8: Modelo em três camadas (mais Clientes Web) do *Webmethods*

Por sua vez, no diagrama seguinte é exemplificada a estrutura genérica e mais usual dos servidores que compõem a suite *webMethods* e também a forma como se encontram interligados e como comunicam:

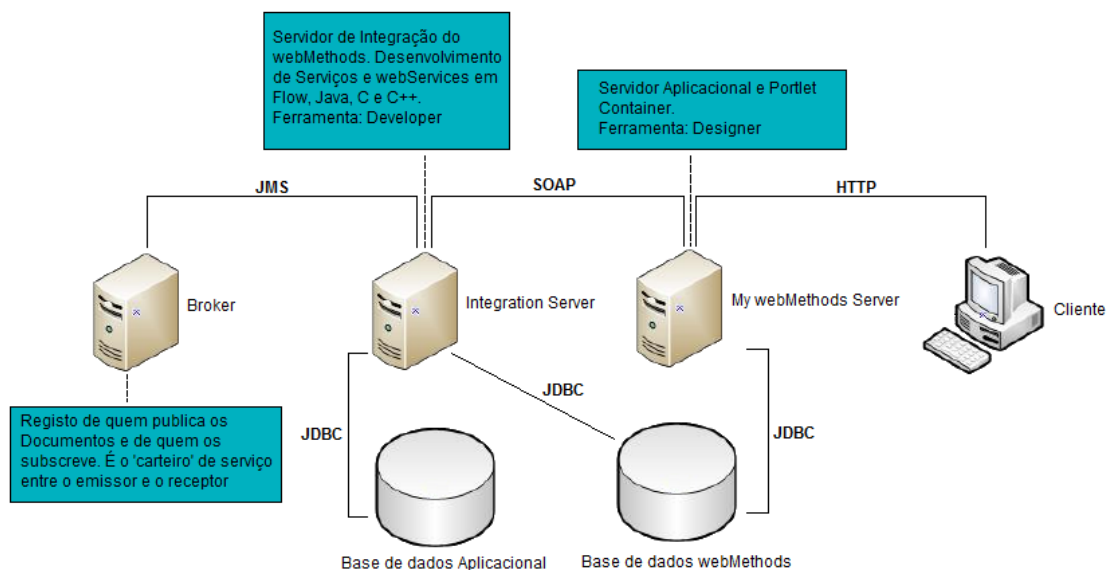


Fig. 9: Arquitectura do *Webmethods*

2.5.1.1 Principais Ferramentas da Plataforma

O *webMethods Designer* é uma ferramenta de desenvolvimento gráfico que teve por base o *Eclipse* e que possibilita a construção de interfaces (aplicações *portlet*) usando a tecnologia *CAF* que permite a construção das ditas *portlets* tendo acesso a dados provenientes de vários sistemas independentes. A construção de várias *portlets*, usando esta ferramenta, compõem uma aplicação final que correrá num servidor específico, o *MyWebMethods Server*.

A tecnologia *CAF* é considerada uma extensão da tecnologia *JSF* [22] [23] (tecnologia que permite a implementação de ecrãs em XML para posteriormente serem interpretados em HTML), que por sua vez é um *Framework MVC* para o desenvolvimento de aplicações Web visuais onde a construção da interface é feita numa zona de edição para a qual se arrastam as diferentes componentes visuais disponibilizadas.

Alguns dos controlos *CAF* recorrem à tecnologia *Ajax* permitindo uma interação assíncrona com o servidor.

Para além de componentes gráficas, no *webMethods Designer* também faculta a modelação de processos e tarefas.

A publicação dos ecrãs e dos processos (incluindo as tarefas humanas) para o *MyWebMethods Server* é executado também através no próprio *webMethods Designer*.

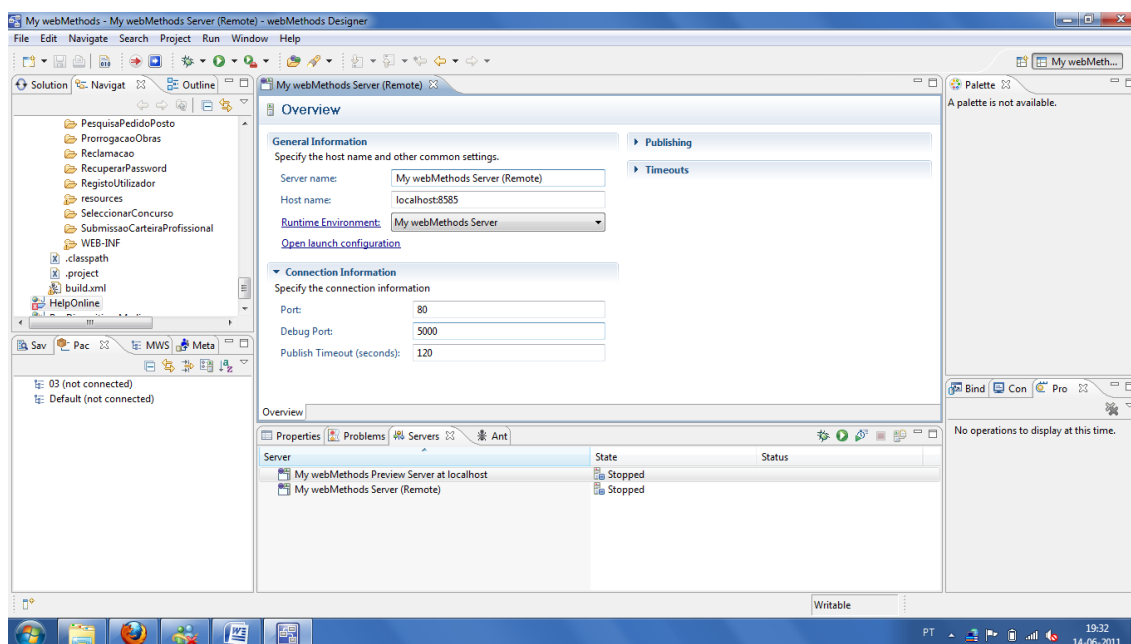


Fig. 10: Interface de desenvolvimento do *Webmethods Designer*

O *MyWebMethods Server* é um servidor aplicacional, baseado em *Jetty*, que possibilita aos utilizadores executarem funções nas componentes *webMethods* e onde os Administradores podem editar/aceder a essas funcionalidades. O servidor *My webMethods* fornece um conjunto amplo de ferramentas administrativas para gerir as funcionalidades desenvolvidas. Este servidor reconhece dois tipos de administradores, com base nas funções que realizam:

- *My webMethods Administrator*: É o Administrador por defeito e pode gerir funções/interfaces disponibilizadas aos utilizadores, assim como outros aspectos de mais alto nível;
- *System Administrator*: Pode efectuar funções de análise, configuração de conteúdos e podem também gerir funções que são disponibilizadas aos utilizadores, tudo isto num nível mais editável que o primeiro;

Sendo assim, a existência deste servidor desenrola-se em torno de duas áreas funcionais:

- *CAF (Composite Application Framework)*, onde se permite desenvolver *portlets* ou mini aplicações web. Estas *portlets* podem ser colocadas em

páginas para possibilitar uma interface ao utilizador. Estas páginas são desenvolvidas numa ferramenta chamada *webMethods Designer*, que mais não é que uma versão do *IDE Eclipse* com um grande conjunto de *plugins* e uma paleta própria de desenvolvimento;

- *Administration*, representada através de páginas de configuração que permitem configurar grande parte daquilo que se desenvolveu.

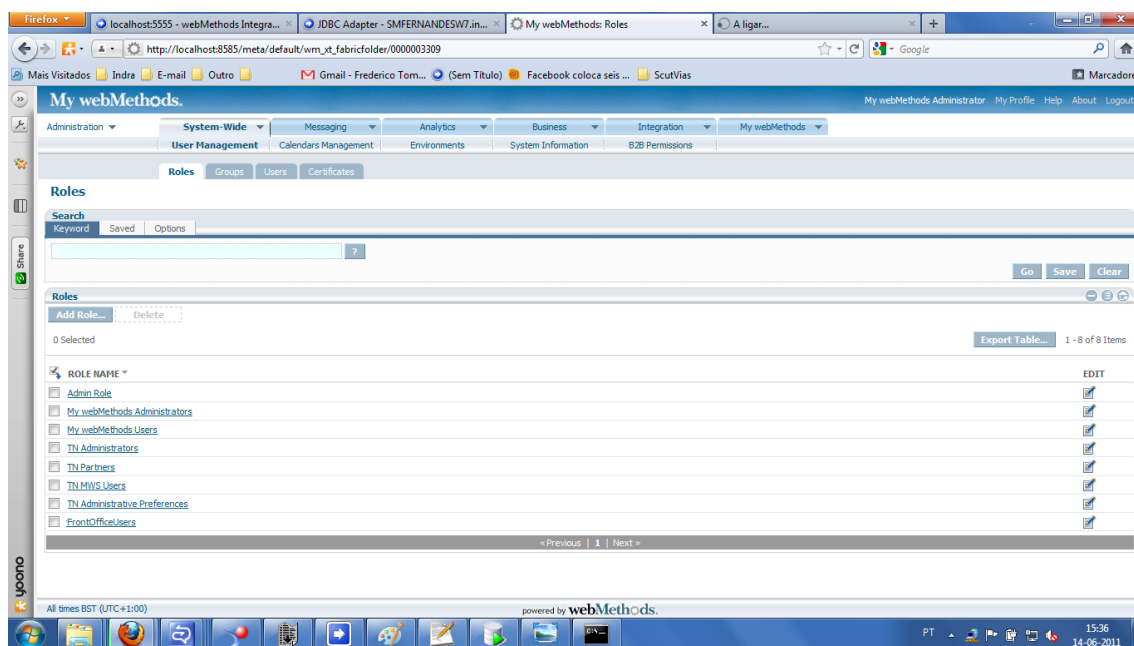


Fig. 11: Interface Web do *MyWebmethods Server*

O *webMethods Developer* é uma ferramenta *IDE* na qual se desenvolvem e testam *webservices*, maioritariamente através da linguagem *Flow* mas também através de outras linguagens de programação (como **Java** e **C**). O uso desses *webservices* assenta no protocolo *SOAP* [24], que é utilizado para troca de informações estruturadas em sistemas distribuídos, recorrendo também a outros protocolos como é o caso do *RPC* ou *HTTP*

O formato das mensagens trocadas através do protocolo *SOAP* é baseado em *XML*, garantindo assim a circulação de dados entre sistemas numa linguagem normalizada.

O *Flow* é uma linguagem de programação visual onde diferentes componentes (que simbolizam código de programação ou mapeamento de variáveis) podem ser arrastados formando uma sequência lógica para realizar determinada tarefa.

Por sua vez, o *Integration Server* (ver Fig. 13) disponibiliza uma interface que permite a configuração de várias componentes que fazem parte do servidor, componentes essas que podem ser pacotes (onde são guardados *webservices*, p.e.), *scheduler*, conexões, entre outros.

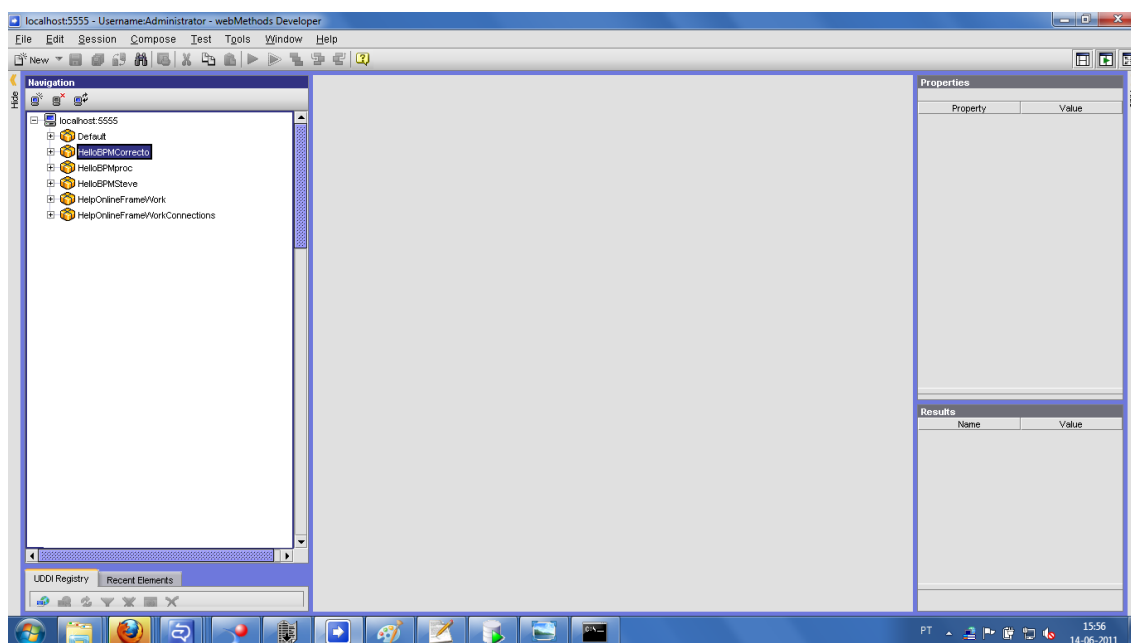


Fig. 12: Interface de desenvolvimento do *Webmethods Developer*

O *Integration Server* é o servidor aplicativo que nos permite integrar os serviços que irão compor o nosso sistema, sendo também ele responsável por executar toda a lógica de negócio necessária ao processamento dos diversos pedidos. É um servidor baseado em Java, e entre os diversos serviços que suporta, encontram-se o mapeamento de dados entre diferentes formatos e a comunicação entre os diferentes sistemas da aplicação. Os serviços desenvolvidos no IS são desenvolvidos majoritariamente na linguagem Java ou então numa linguagem de programação gráfica, própria da suite *webMethods*, denominada *Flow*. A ferramenta de desenvolvimento de serviços fornecida pelo *webMethods* é o *webMethods Developer*. A administração, configuração do IS é feita através de uma página Web que este disponibiliza.

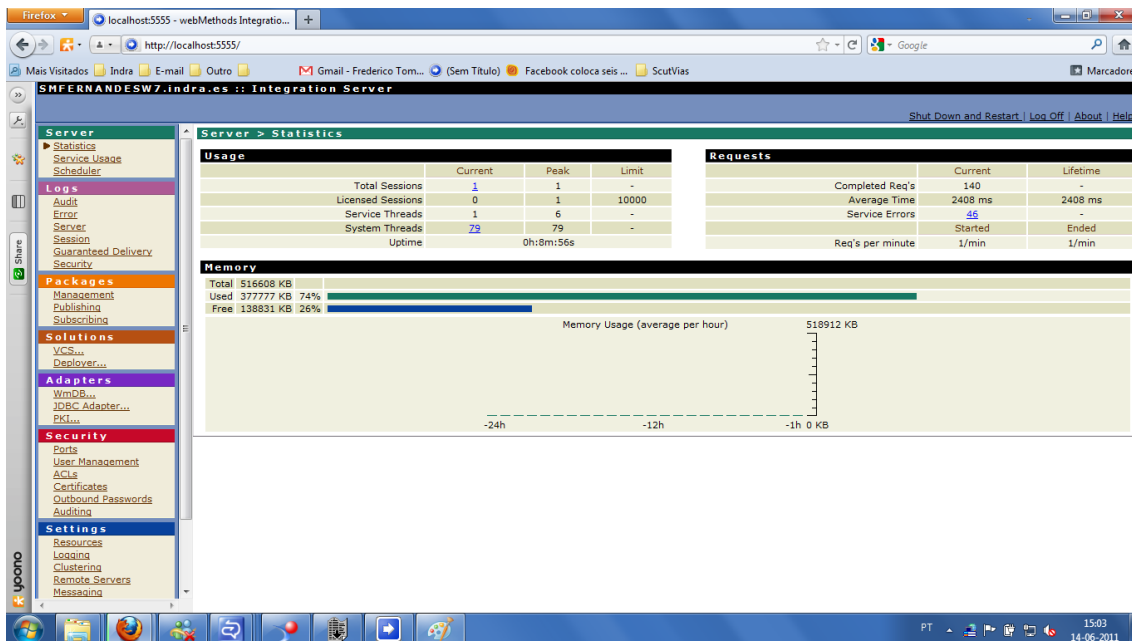


Fig. 13: Interface Web do *Integration Server*

O *Broker* (ver Fig. 9) é o servidor que regista quem publica os documentos e quem os subscreve, mantendo um registo dos emissores e receptores, e agindo com a função de “carteiro” de documentos com, por exemplo, o *Integration Server*. Numa arquitectura mais simples o emissor e o receptor será o mesmo, o nosso caso o *Integration Server*, mas numa arquitectura de maior dimensão, poderá fazer a ligação entre diversos *Integrations Servers*.

2.5.2 Arquitectura lógica

A imagem abaixo representa a arquitectura lógica que dá suporte às aplicações que se desenvolveram.

Como é visível, existe uma possível interacção entre os Utilizadores e uma interface desenvolvida na tecnologia *CAF* da suite *webMethods*, sendo este processo feito através do *MyWebMethods Server*. Para acesso à informação contida na base de dados aplicacional, a camada da *User Interface* fará chamadas a serviços disponibilizados no *Integration Server* no formato *SOAP* sobre um canal de comunicação *HTTP*. Os serviços a desenvolver no *Integration Server* serão implementados numa de duas formas, conforme a complexidade da sua lógica interna:

- Em *Flow* (orientação da suite *WebMethods*) serão desenvolvidos os serviços cuja lógica seja simples de implementar;
- Em *Java* serão implementados os serviços mais complexos.

Todos os serviços a desenvolver poderão ficar disponíveis para o Cliente, sendo possível, caso necessário, expor os mesmos a aplicações externas, através de *webservices*. Desta forma, respeita-se a arquitectura *SOA* e recomenda-se a reutilização de serviços.

A persistência dos dados aplicativos será efectuada com recurso a um esquema Oracle que será consultado através de serviços implementados no IS.

A integração com as aplicações já existentes no Cliente será realizada através de chamadas a serviços disponíveis *no Integration Server*. Estas chamadas, sendo internas, são lançadas no próprio servidor, sem recurso a outras tecnologias.

A interação com a gestão documental e uma *gateway* de pagamentos é feita com o consumo de *webservices* que estas plataformas expõem. Estas chamadas são feitas usando o formato *SOAP* sobre um canal de comunicação *HTTP*.

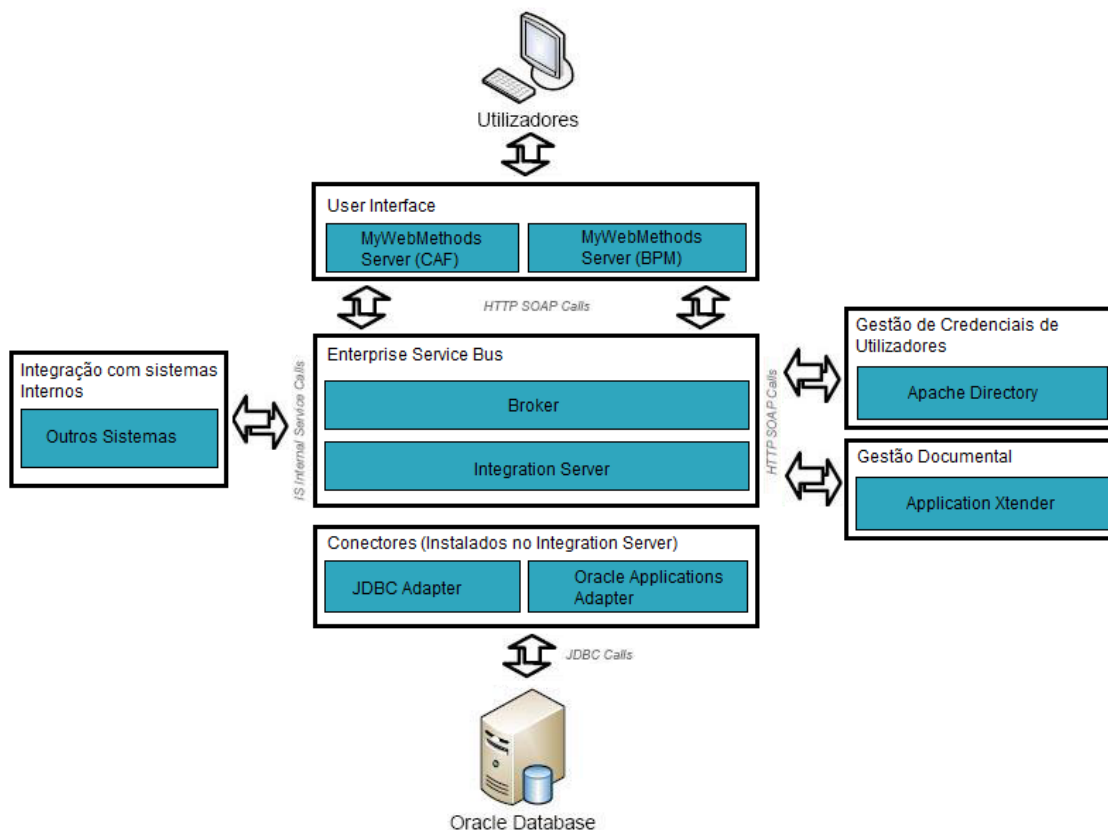


Fig. 14: Arquitectura Lógica

2.5.3 Arquitectura Física

A arquitectura física que importa referir é aquela que dará suporte à aplicação em ambiente de produção. Dado que a tecnologia a usar é *WebMethods* e estando já instalado no Cliente uma solução para esta necessidade e que suportava todos os sistemas, foi usada a mesma arquitectura, que se espelha a título de exemplo na imagem abaixo e que é usada para a equipa Indra no Cliente. Importa referir também que toda a arquitectura se baseia naquela que é imposta pela utilização da suite *webMethods*, podendo apenas ser escalável, consoante a necessidade do cliente em questão.

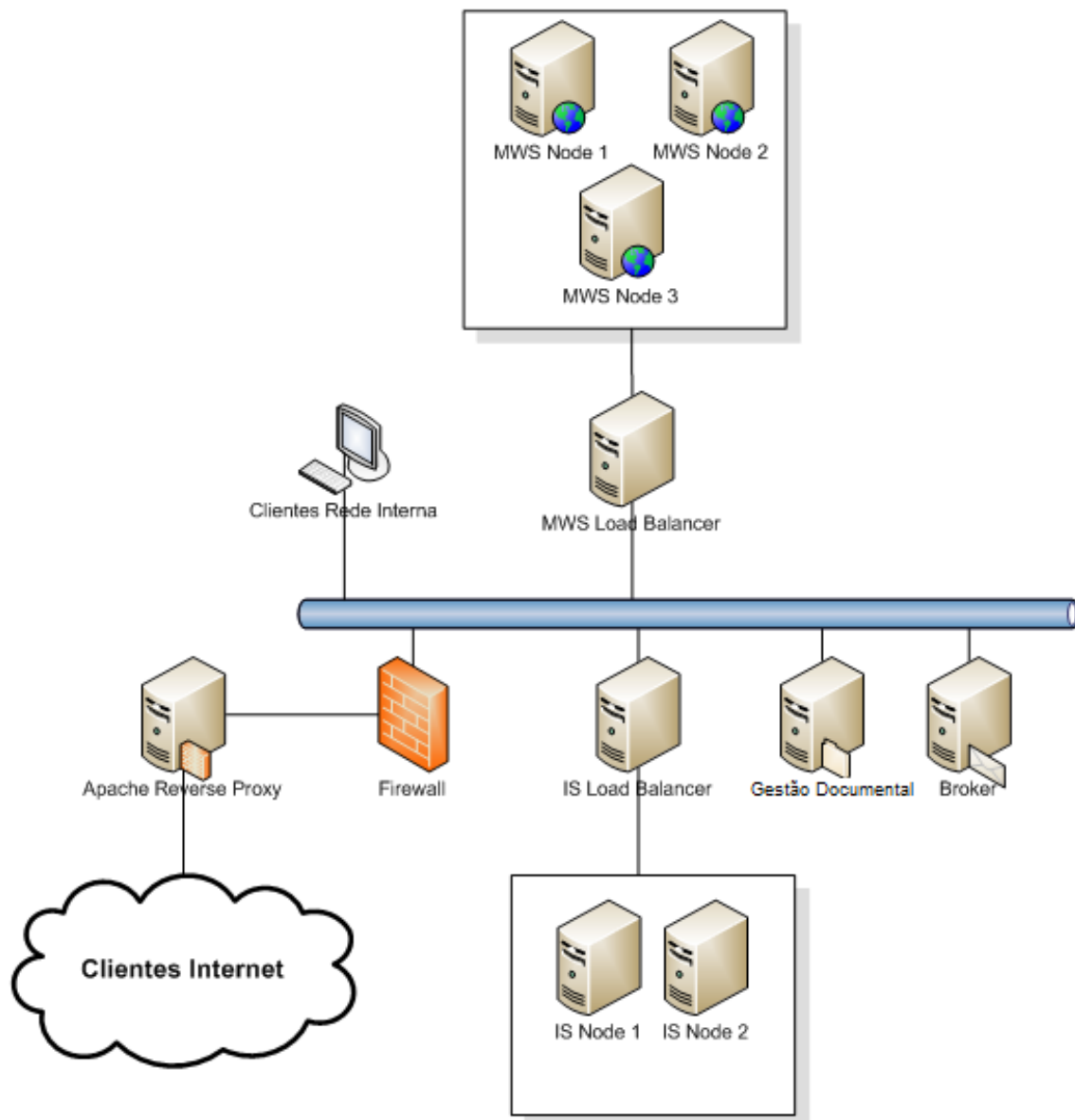


Fig. 15: Exemplo de Arquitectura Física

2.5.4 Ferramentas auxiliares

Base de dados

Foram usadas ferramentas de base de dados que suportam a informação e os dados disponíveis. O *SGBD* usado em ambiente de produção foi o Oracle 9i pois já era usado em grande parte das aplicações existentes no cliente, enquanto que em ambiente de desenvolvimento foi usado o Oracle 11g.

A ferramenta cliente de Base de Dados mais usada foi o *SQL Developer* [3], também da Oracle.

Baretail [1]

Esta aplicação foi utilizada para visualizar os ficheiros de *log*, neste caso dos servidores MWS e IS em tempo real.

Uma das suas funções mais interessantes é o *Highlighting*, que permite aplicar cores de fundo a linhas que contenham uma determinada expressão, o que por vezes é muito útil na detecção de erros.

TortoiseSVN [2]

Este foi o cliente de *SVN* usado e que permitiu o controlo de versões dos vários ficheiros utilizados pela equipa de desenvolvimento. Revelou-se uma das ferramentas mais utilizadas e facilitou o trabalho corporativo.

Capítulo 3

Trabalho realizado

3.1 Gestão de actividades

O Planeamento previsto no início do estágio não foi cumprido de forma rigorosa em resultado de vários factores que foram surgindo, como por exemplo:

- Pedido do cliente para adição de novas funcionalidades;
- Alteração na estratégia comercial;

Apesar da discrepância existente, importa referir que todas as funcionalidades planeadas para o *Help Online* foram totalmente implementadas.

3.1.1 Calendarização

	①	Nome	Duração	Início	Fim	Predecessores
1		Integra-t - Evento de acolhimento aos novos colaboradores	5d?	06/09/2010	10/09/2010	
2		<input type="checkbox"/> Integração no Projecto	15d?	13/09/2010	01/10/2010	1
3		Integração na sede do Cliente	1d?	13/09/2010	13/09/2010	
4		Leitura de Documentos dos projectos a decorrer	4d?	14/09/2010	17/09/2010	3
5		<input type="checkbox"/> Integração tecnológica	10d?	20/09/2010	01/10/2010	4
6		Formação Introdutória	3d?	20/09/2010	22/09/2010	
7		webMethods Developer	1d?	23/09/2010	23/09/2010	6
8		webMethods Designer	1d?	24/09/2010	24/09/2010	7
9		Tutoriais	5d?	27/09/2010	01/10/2010	8
10		<input type="checkbox"/> Estudo e Visibilidade do Sistema (EVS)	5d?	04/10/2010	08/10/2010	2
11		Reunião	1d?	04/10/2010	04/10/2010	
12		Análise das necessidades do Cliente	1d?	04/10/2010	04/10/2010	
13		Avaliação do tempo necessário	1d?	05/10/2010	05/10/2010	12
14		Realização do planeamento	2d?	06/10/2010	07/10/2010	13
15		Reunião e Revisão	1d?	08/10/2010	08/10/2010	14
16		<input type="checkbox"/> Análise do Sistema de Informação (ASI)	18d?	11/10/2010	03/11/2010	10
17		Entrevista com stakeholder	1d?	11/10/2010	11/10/2010	
18		<input type="checkbox"/> Início do Documento de Análise Funcional	16d?	12/10/2010	02/11/2010	17
19		Documento de Requisitos	7d?	12/10/2010	20/10/2010	
20		Protótipo de Baixo nível (Especificação de ecrãs)	9d?	21/10/2010	02/11/2010	19
21		Reunião para aprovação da Análise	1d?	03/11/2010	03/11/2010	18
22		<input type="checkbox"/> Desenho do Sistema de Informação	10d?	04/11/2010	17/11/2010	16
23		Reunião com coordenador técnico	1d?	04/11/2010	04/11/2010	
24		<input type="checkbox"/> Documento de Desenho técnico	8d?	05/11/2010	16/11/2010	23
25		Arquitecturas	4d?	05/11/2010	10/11/2010	
26		Modelo de Dados	4d?	11/11/2010	16/11/2010	25
27		Reunião para aprovação do Desenho técnico	1d?	17/11/2010	17/11/2010	24
28		<input type="checkbox"/> Construção do Sistema de Informação (CSI)	34d?	18/11/2010	04/01/2011	22
29		Reunião com o Cliente e Coordenador	1d?	18/11/2010	18/11/2010	
30		Reformular DSI e ASI	1d?	19/11/2010	19/11/2010	29
31		<input type="checkbox"/> Módulo de Gestão de Imagens	3d?	22/11/2010	24/11/2010	30
32		Criação do CAF relativo a este módulo	1d?	22/11/2010	22/11/2010	
33		Criação dos serviços relativos à Gestão de imagens	1d?	22/11/2010	22/11/2010	
34		Upload de imagens para o servidor	1d?	23/11/2010	23/11/2010	32,33
35		Adicionar, Remover	1d?	24/11/2010	24/11/2010	34
36		<input type="checkbox"/> Módulo árvore de Tópicos de Ajuda	2d?	25/11/2010	26/11/2010	31
37		Criação de serviços	1d?	25/11/2010	25/11/2010	
38		Criação da Tree em CAF	1d?	26/11/2010	26/11/2010	37
39		<input type="checkbox"/> Módulo de wysiwyg	4d?	29/11/2010	02/12/2010	36
40		Actualização do Script de imagens consuante servidor	1d?	29/11/2010	29/11/2010	

41		Melhorias wysiwyg	1d?	30/11/2010	30/11/2010	40
42		Criação da Portlet de edição	1d?	01/12/2010	01/12/2010	40,41
43		Criação de serviços	1d?	02/12/2010	02/12/2010	42
44		<input type="checkbox"/> Módulo de Pesquisa Manual	2d?	03/12/2010	06/12/2010	39
45		Ajuste dos serviços para FrontOffice	1d?	03/12/2010	03/12/2010	
46		Ajuste da tree em CAF	1d?	06/12/2010	06/12/2010	45
47		<input type="checkbox"/> Módulo de Pesquisa por palavra-chave	4d?	07/12/2010	10/12/2010	44
48		Criação de serviços	1d?	07/12/2010	07/12/2010	
49		Criação do algoritmo de pesquisa	2d?	08/12/2010	09/12/2010	48
50		Interface onde são mostrados resultados da pesquisa	1d?	10/12/2010	10/12/2010	49
51		<input type="checkbox"/> Módulo de Visualização de tópicos de ajuda	1d?	13/12/2010	13/12/2010	47
52		Criação de Serviços	1d?	13/12/2010	13/12/2010	
53		Testes	10d?	14/12/2010	27/12/2010	51
54		Adaptação do Help Online a um projecto	1d?	28/12/2010	28/12/2010	53
55		Reunião com o Cliente	1d?	29/12/2010	29/12/2010	54
56		Reformular ASI e DSI	4d?	30/12/2010	04/01/2011	55
57		<input type="checkbox"/> Resolução de problemas identificados na ultima reunião	29d?	05/01/2011	14/02/2011	28
58		Solucionar problemas com passagem de informação entre views distintas	1d?	05/01/2011	05/01/2011	
59		Colocar o layout gráfico do WYSIWYG igual ao utilizado no projecto,	1d?	06/01/2011	06/01/2011	58
60		Colocar a associação de imagens ao nível do projecto.	1d?	07/01/2011	07/01/2011	59
61		Adicionar pesquisa de imagens na edição de conteúdos (Edit),	2d?	10/01/2011	11/01/2011	60
62		Adicionar pesquisa de imagens na gestão de imagens (BackOffice)	1d?	12/01/2011	12/01/2011	61
63		Possibilitar escolha de visualização do tópico de ajuda.	1d?	13/01/2011	13/01/2011	62
64		Possibilitar Títulos/Conteúdos Multilanguage (Utilizar o Default Locale)	1d?	14/01/2011	14/01/2011	63
65		Possibilitar desactivação de conteúdos, reflectindo-se em FrontOffice	1d?	17/01/2011	17/01/2011	64
66		Desabilitar o menu lateral do projecto ao abrir os tópicos de ajuda do Help	1d?	18/01/2011	18/01/2011	65
67		Na edição de conteúdos mostrar todos os tópicos antecedentes.	1d?	19/01/2011	19/01/2011	66
68		Construção da Interface de Administração	10d?	20/01/2011	02/02/2011	67
69		Visualizar imagem numa página própria	1d?	03/02/2011	03/02/2011	68
70		Modificar a mensagem no caso do conteúdo estar desactivado	1d?	04/02/2011	04/02/2011	69
71		Conteúdo inactivo por default; Modificar a pesquisa por palavra-chave	1d?	07/02/2011	07/02/2011	70
72		Upload de imagens para servidor http;	2d?	08/02/2011	09/02/2011	71
73		Criar um Histórico para o Help Online;	2d?	10/02/2011	11/02/2011	72
74		Actualização de ASI e DSI	1d?	14/02/2011	14/02/2011	73
75		Testes	15d?	15/02/2011	07/03/2011	57
76		Relatório Preliminar	15d?	15/02/2011	07/03/2011	57
77		<input type="checkbox"/> Implementação e Aceitação do Sistema (IAS)	18d?	08/03/2011	31/03/2011	75
78		Manual de Instalação	4d?	08/03/2011	11/03/2011	
79		Criação dos Links de acesso Help Online -> Projecto	3d?	14/03/2011	16/03/2011	78
80		Criação dos Conteúdos de ajuda do Projecto	7d?	17/03/2011	25/03/2011	79
81		Actualização de documentos	4d?	28/03/2011	31/03/2011	80
82		Semana de formação - IBM Infosphere	5d?	01/04/2011	07/04/2011	77
83		Relatório Final	45d?	08/04/2011	09/06/2011	82

3.2 Estudo de viabilidade do sistema (EVS)

Nesta etapa foram analisadas quais as necessidades que o cliente tinha para dar seguimento ao projecto do *Help Online*. Nesta fase foi-me atribuída pela Gestora de Projecto a responsabilidade da realização do *Help Online*. Ficou também definido que este projecto seria o tema central da tese de mestrado o que implicaria portanto ter um prazo máximo de 9 meses.

Visto o cliente ser bastante flexível e existir um contacto próximo com a empresa, foi agendada uma reunião para dar a conhecer o projecto ao cliente.

A reunião focou os seguintes tópicos:

- **O que será o *Help Online*:** Começou por ser fazer uma demonstração das diferenças relativas ao que já existia feito, e ao que poderíamos fazer;
- **Requisitos:** Foram clarificados os requisitos que o cliente já tinha imposto sobre a aplicação anterior, requisitos esses que não eram cumpridos. Falou-se nomeadamente do facto de o conteúdo de ajuda não poder ser editável;
- **O *BackOffice*:** Foram mostradas as principais funcionalidades a serem propostas para um utilizador de *BackOffice*;
- **O *FrontOffice*:** Foram mostradas as principais funcionalidades a serem propostas para um utilizador de *FrontOffice*;
- **Possível Modelo de Dados:** Demonstração do Modelo de dados base que suportaria a área de negócio;

Devido à boa relação mantida entre o Cliente e a empresa, esta etapa não foi muito aprofundada pois já tinha sido discutido previamente a ideia de se construir um novo sistema de *Help Online*. Esta etapa baseou-se então na apresentação feita ao cliente, na qual os tópicos abordados foram aceites entre ambas as partes.

3.3 Análise do sistema de informação (ASI)

Nesta etapa foi feita uma discriminação detalhada do *Help Online* através de um documento de especificação funcional.

Esse documento conta com uma descrição pormenorizada da aplicação, dizendo qual o seu âmbito assim como também são mostrados todos os requisitos inerentes.

São apresentadas as Especificações de ecrãs.

Finalizado o Documento, foi necessário proceder à sua aprovação por parte da equipa funcional.

3.3.1 Requisitos Funcionais

Estes requisitos foram obtidos através de um trabalho de análise com base em documentação interna já existente, reuniões realizadas junto com o cliente e também através de conhecimento partilhado dentro da equipa.

A listagem foi feita com a seguinte estrutura:

- Código requisito: Código a ser usado por qualquer referência que se faça a este requisito específico;
- Descrição Requisito: Descrição do requisito em questão;
- Dependências: Uma ou mais dependências a requisitos anteriores;

De seguida são mostrados alguns exemplos, ficando a lista com todos os requisitos funcionais em anexo:

Código Requisito	Descrição Requisito	Dependências
HO_001	Toda a informação é organizada em tópicos de ajuda.	
HO_004	Os tópicos de ajuda deverão ter a possibilidade de ser editados por um técnico que apresente um perfil adequado.	
HO_008	O utilizador a editar um tópico de ajuda apenas poderá modificar o seu título, estado e conteúdo.	
HO_014	Cada tópico de ajuda tem associados apenas um título e um conteúdo. Esses títulos e conteúdos por sua vez podem ser representados em várias línguas.	

Tabela 6: Exemplo de Requisitos Funcionais na estrutura do documento

Número Total de Requisitos Funcionais: 27

3.3.2 Diagramas Caso de Uso

A metodologia usada pela empresa não implicou a criação dos diagramas de caso de uso, sendo este trabalho feito por vontade própria e porque a experiência académica assim o indicou. Foram criados de forma a descrever elaboradamente os Casos de Uso com o objectivo de refinar a análise de requisitos funcionais da aplicação *Help Online*.

As Figura 16, 17 e 18 representam os diagramas de Casos de Uso e demonstram as interações dos vários utilizadores com o sistema.

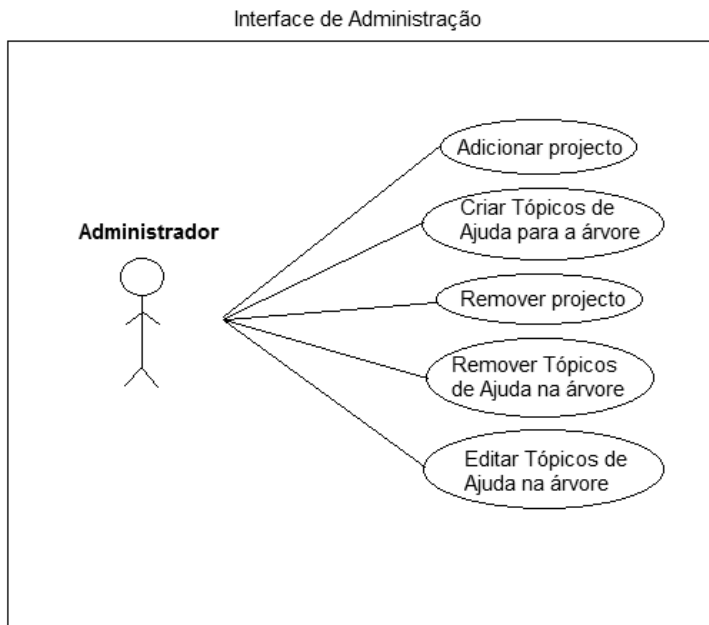


Fig. 16: Caso de Uso – Interface de administração

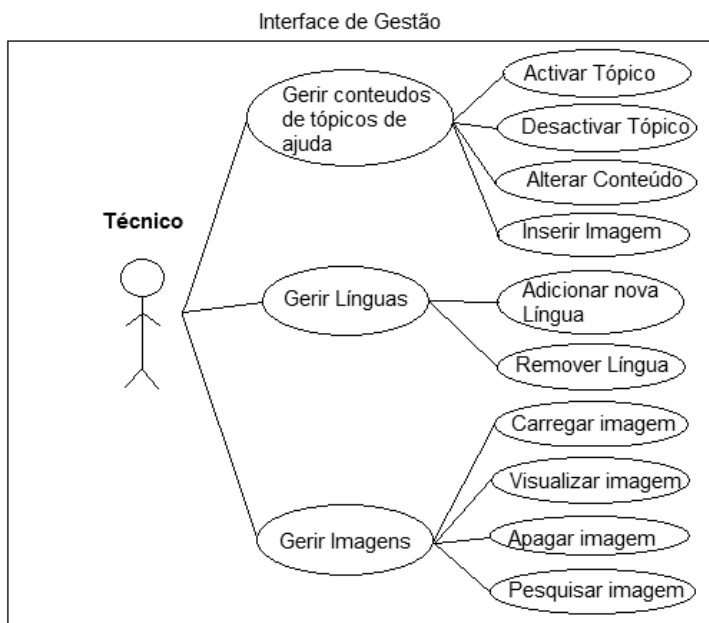


Fig. 17: Caso de Uso – Interface de gestão

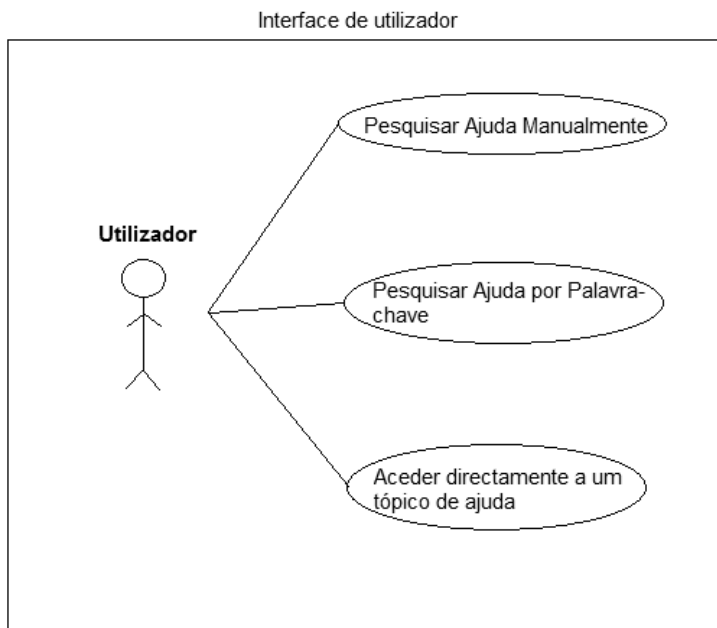


Fig. 18: Caso de Uso – Interface de utilizador

Para além dos diagramas foi também feita uma descrição elaborada de cada caso de uso, permitindo assim compreender o fluxo de actividade e interações entre o utilizador e o sistema. Devido ao enorme espaço que a descrição de todos os casos de uso ocuparia, são apenas apresentados alguns exemplos para demonstrar a linha orientadora seguida:

Caso de Uso: Remover projecto
Actor Principal: Administrador
Pré-Condições: Tem que existir um projecto
Pós-Condições: O projecto é apagado
Cenário Principal: <ol style="list-style-type: none"> 1. O Administrador entra na página de administração; 2. Dentro do subgrupo ‘Gestão de Projectos’ terá que seleccionar o projecto pretendido; 3. Clica no botão ‘Remover Projecto’; 4. É despoletado um <i>Popup</i> de confirmação; 5. Caso tenha a certeza, o administrador clica em ‘OK’;
Cenários Alternativos: <ol style="list-style-type: none"> 5 a) Caso o administrador clique em ‘Cancelar’. <ol style="list-style-type: none"> 1. Caso de uso termina;

Tabela 7: Exemplo de descrição de Caso de Uso

Caso de Uso: Activar Tópico
Actor Principal: Técnico
Pré-Condições: Tem que existir um projecto e um Tópico desactivo
Pós-Condições: O tópico fica activo e visível para o utilizador
Cenário Principal: <ol style="list-style-type: none"> 1. O técnico entra na interface de Gestão; 2. Dentro do subgrupo 'Árvore de Conteúdos' terá que seleccionar o tópico pretendido; 3. Será aberta a interface de edição de tópicos de ajuda; 4. Dentro do subgrupo 'Conteúdo' seleccionar 'Activo'; 5. Clicar em 'Terminar';
Cenários Alternativos: <ol style="list-style-type: none"> 5 a) Caso o administrador clique em 'Cancelar'. <ol style="list-style-type: none"> 1. Caso de uso termina;

Tabela 8: Exemplo de descrição de Caso de Uso

3.3.3 Requisitos Não Funcionais

O sistema tem de possuir determinadas características que irão contribuir para o desempenho global do sistema. Esses requisitos não funcionais foram descritos segundo a mesma estrutura:

Código Requisito	Descrição Requisito	Dependências
HO_NF_002	A edição dos conteúdos do <i>Help Online</i> deverá ser efectuada de forma rápida e intuitiva, minimizando eventuais dúvidas de processamento de texto e inserção/eliminação de conteúdo ilustrativo.	
HO_NF_005	O <i>Help Online</i> deverá ser uma funcionalidade dinâmica, tendo capacidade de ser adaptar a distintas estruturas aplicacionais.	

Tabela 9: Exemplos de Requisitos Não Funcionais na estrutura do documento

Em suma, a listagem elaborada foi de encontro a:

- **Usabilidade:** O GUI deve ter uma curva de aprendizagem curta, reutilizando os modelos mentais dos utilizadores [4] [5].
- **Fiabilidade:** Não devem ser permitidas excepções não controladas.
- **Adaptabilidade:** A aplicação terá que se adaptar facilmente a vários ambientes onde seja presumível funcionar.

Número total de Requisitos Não Funcionais: 6

3.3.4 Especificação de Ecrã

Foi necessário efectuar um primeiro esboço do que seriam as futuras implementações. Este esboço foi inserido na secção de especificação de ecrãs e é importante na medida que serve para ter uma visão geral de como efectuar a implementação, sendo também possível verificar problemas técnicos que possam surgir no futuro.

Foi decidido que os protótipos de baixa fidelidade seriam feitos no próprio *Webmethods Designer*, arrastando as várias componentes e formando assim apenas uma interface estática mas que por sua vez nos pouparia trabalho na fase de implementação.

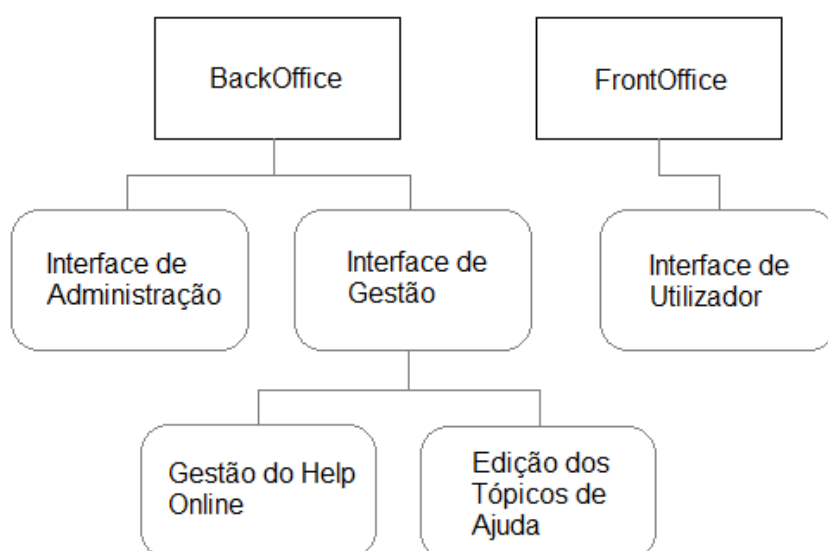
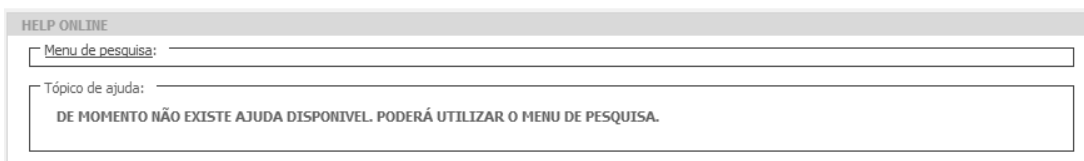


Fig. 19: Ecrãs do sistema *Help Online*

A estrutura de ecrãs, representada na Fig. 19, foi dividida em 2 subgrupos: BackOffice e FrontOffice. O primeiro pressupõe que a utilização do ecrã será feita por alguém com perfil de administração do sistema *Help Online*. O segundo pressupõe que a utilização do ecrã é feita por alguém que só pretende utilizar o sistema de ajuda. De seguida é exemplificada a estrutura de uma especificação de ecrã:

1. Ecrã de Consulta de Tópicos de Ajuda



HELP ONLINE

Menu de pesquisa: _____

Tópico de ajuda: _____

DE MOMENTO NÃO EXISTE AJUDA DISPONIVEL. PODERÁ UTILIZAR O MENU DE PESQUISA.

2. Descrição

Descrição das Funcionalidades do ecrã.




3. Campos

Campo	Tipo	O/F	Descrição	Origem
Árvore de Tópicos	<i>Toogle Tree</i>	F		

4. Regras de negócio

Campo	Regra de Negócio
Árvore de Conteúdos Vazia	Numa árvore de conteúdos vazia irá constará exclusivamente a sua raiz, sendo que os tópicos quando forem criados serão subtópicos deste nível.
Inexistência de tópico de ajuda que contenha palavra-chave	O utilizador é informado que não existe nenhum tópico que contenha a palavra-chave usada

5. Acções

Acção	Descrição
	Permite ocultar e mostrar a secção de pesquisa de conteúdos.
	Permite seleccionar o separador onde se encontra a árvore para fazer uma pesquisa manual.
	Permite expandir ou abrir os ramos da árvore, sendo possível passar a visualizar o seu conteúdo.

6. Integrações com outros sistemas

Sistema	Descrição
N/A	N/A

Tabela 10: Exemplo de especificação de um ecrã

3.4 Desenho do sistema de informação (DSI)

Nesta etapa houve necessidade de demonstrar ao cliente o processo de desenvolvimento a ser usado. Foi feita uma descrição das camadas lógicas seguindo-se um pequeno resumo sobre a arquitectura na qual assentou o projecto.

Foi feito também os Modelo de Dado, finalizando-se com uma descrição dos principais serviços a serem realizados.

Por fim, assim como aconteceu no documento funcional, o documento proveniente desta etapa também requereu uma aprovação.

3.4.1 Modelo Relacional [6]

O Modelo Relacional abaixo permite criar uma representação lógica consistente da informação na base de dados.

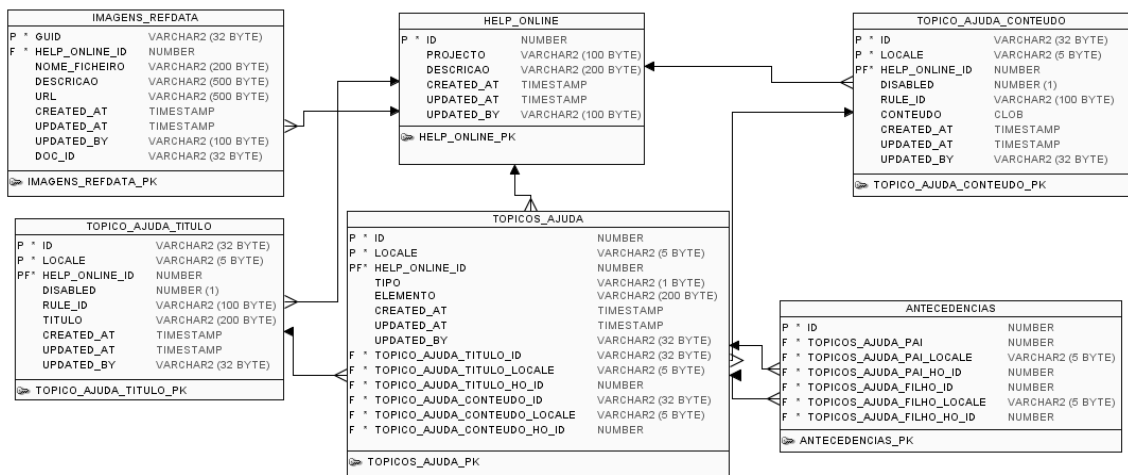


Fig. 20: Modelo Relacional de Dados

3.4.2 Exemplo de Descrição de Serviço

Nesta etapa foi também efectuada uma listagem com os serviços a serem desenvolvidos. Para cada serviço especificou-se quais eram os seus inputs, outputs assim como uma pequena descrição da operação que viria a realizar. Foi também tido em conta o facto de o serviço em questão poder utilizar outros serviços já existentes, dai

ter sido adicionado o campo 'Dependências'. A descrição que contempla todos os serviços esta em anexo, porem de seguida encontram-se dois exemplos do que foi desenvolvido:

Nome: criarProjecto
Inputs: nome, descrição, <i>default_language</i> , <i>updated_by</i> ; Outputs: status; Dependências: N/A; Descrição: Cria a estrutura na base de dados necessária para começar a introduzir tópicos de ajuda;

Tabela 11: Exemplo de Descrição do serviço *criarProjecto*

Nome: criarTopicoAjuda
Inputs: id_projecto, <i>loc</i> , tipo, <i>portlet</i> , elemento, <i>updated_by</i> , titulo, accao_conteudo. id_conteudo, descricao_conteudo, id_antecedente. Outputs: status. Dependências: N/A. Descrição: Criar um tópico de ajuda com um conteúdo vazio;

Tabela 12: Exemplo de Descrição do serviço *criarTopicoAjuda*

3.5 Construção do sistema de informação (CSI)

Nesta fase procedeu-se à construção dos diferentes componentes do sistema de informação, a partir do conjunto de especificações lógicas e físicas feitas nas etapas anteriores.

Para tal foram usados essencialmente três ambientes:

- Ambiente de Desenvolvimento Local;
- Ambiente de Desenvolvimento/Testes de integração;
- Ambiente de Testes;

O Ambiente Local serviu para a resolução de pequenos problemas pois era à partida o ambiente menos carregado de dados e conseqüente o mais rápido para se fazer a publicação da nossa aplicação.

No Ambiente de desenvolvimento estavam presentes os projectos a desenvolver na altura, incluído a nossa aplicação. Neste ambiente era possível montar o nosso Help Online sobre um projecto do cliente.

O Ambiente de testes replicava o que existia no ambiente de desenvolvimento com a vantagem dos dados provenientes da BD serem mais coerentes. Este servidor era usado maioritariamente pela equipa funcional.

De seguida é elaborada uma explicação das ferramentas segundo o modelo de três camadas [7]:

Camada de Apresentação;

Camada de Lógica de Negócio;

Camada de Dados;

Por fim é descrita a fase de testes e qualidade do sistema *Help Online*, passando pela realização dos mesmos e finalizando na avaliação dos resultados obtidos.

3.5.1 Camada de Apresentação

Para o desenvolvimento da camada de apresentação foi usada a ferramenta *Webmethods Designer*, a qual me permitiu criar as interfaces (Fig. 19) fazendo essencialmente *drag-and-drop* das várias componentes presentes na paleta da aplicação.

Grande parte do código Java por trás da interface é gerado automaticamente pela aplicação embora seja necessária a criação de vários métodos ‘auxiliares’.

Foram estas as minhas maiores preocupações [8] durante a construção das interfaces:

- **Consistência:** Visto as componentes da interface serem já predefinidas pela ferramenta de desenvolvimento, esta característica não requereu grandes preocupações.
- **Feedback informativo:** Todas as acções a serem realizadas pelo utilizador possuem uma resposta do sistema informativa, quer positiva quer negativa
- **Prevenção e tratamento de erros:** Foram prevenidos todos os possíveis erros na medida do possível. Esta fase porém foi sendo melhorada e só foi terminada depois da fase de testes.
- **Reversão de acções:** O utilizador poderá sempre que quiser fazer uma reversão das suas acções. Exemplo disto são situações em que uma dada tarefa é feita em vários passos, podendo o utilizador sempre que desejar voltar ao passo anterior.
- **Baixa carga de memorização:** A interface foi feita de forma a estar bem estruturada e existir um equilíbrio que permita ao utilizador relacionar elementos com determinadas acções.

Total de Interfaces: 4

Total aproximado de Linhas de código: 3508

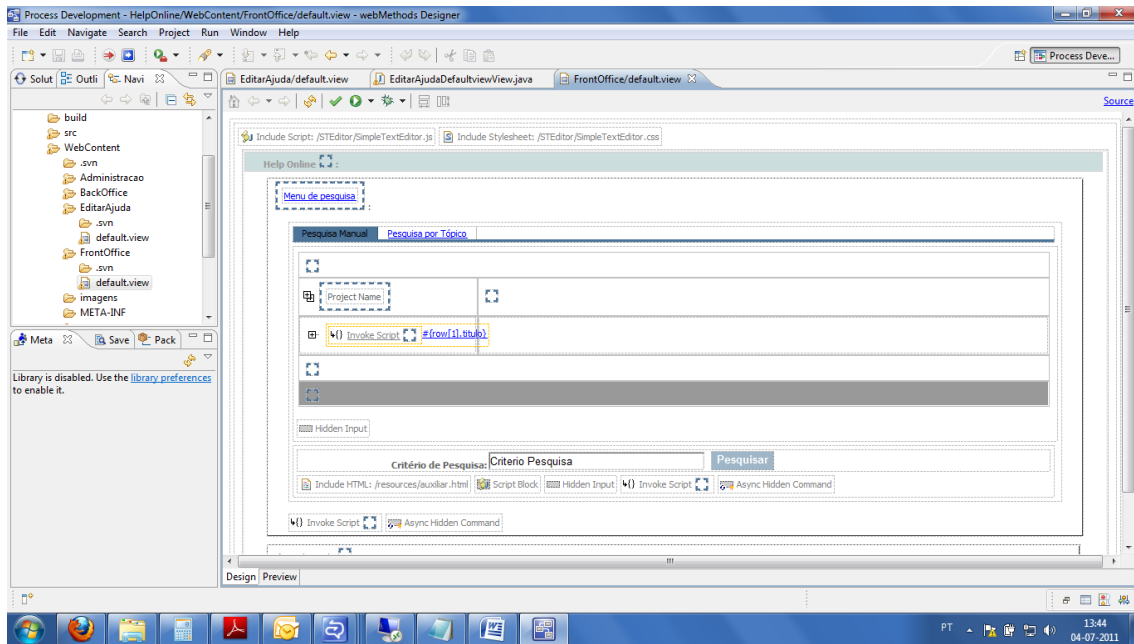


Fig. 21: Exemplo de construção de interface no Webmethods Designer

Devido à enorme quantidade de código existente, apenas irei destacar um método elaborado, cujo objecto é construir a árvore de tópicos de ajuda.

```

import com.webmethods.caf.backoffice.Topico;

private Topico_ajuda[] array_topicos, array_topicos_criterio;
private void buildTree() {
    getGetTopicos2().refresh();

    __topicos_outputs[] lista_topicos =
getGetTopicos2().getResult().getTopicos_outputs();

    array_topicos = new Topico[lista_topicos.length];

    Topico topicoActual = new Topico();
    Topico topicoActualAux = new Topico();

    //preencher array de topicos
    for(int i = 0; i < lista_topicos.length; i++){
        if(i == lista_topicos.length - 1)
            array_topicos[i] = new
Topico(Integer.parseInt(lista_topicos[i].getId().toString()),lista_top
icos[i].getTitulo(),0);

        else
            array_topicos[i] = new
Topico(Integer.parseInt(lista_topicos[i].getId().toString()),lista_top
icos[i].getTitulo(),Integer.parseInt(lista_topicos[i].getAntecedente()
.toString()));
    }
}

```

```

}

for(int i = 0; i < lista_topicos.length; i++){
    topicoActual = array_topicos[i];
    if(topicoActual.getAntecedente() == 1){
        treeLeafs.add(new Object[] {null, topicoActual});
    }else{
        for(int a = 0; a < lista_topicos.length; a++){
            topicoActualAux = array_topicos[a];
            if(topicoActualAux.getId() ==
topicoActual.getAntecedente()){
                treeLeafs.add(new Object[]
{topicoActualAux, topicoActual});
            }
        }
    }
}

tree = new ListTreeContentProvider(treeLeafs, "#{row[1]}",
"#{row[0]}", null);
}

```

Tabela 13: Exemplo de Código escrito em Java

Principais linhas orientadoras a seguir:

- Código [11] [12]
 - **Comentários:** código deve ser devidamente documentado para que seja facilmente legível.
 - **Nomes:** Ter atenção aos nomes atribuídos de modo a que identifiquem devidamente o objecto em questão.
 - **Excepções e tratamento de erros:** Tratar devidamente de todas as excepções e erros que poderão surgir ao longo dos testes do projecto, para minimizar os custos de manutenção após a fase de fecho do projecto.
- CAF [11]
 - **Convenção para nomes:** Os *portlets* deverão ser compostos por nomes abreviados, o mais explícito possível.

3.5.2 Camada de Lógica de Negócio

Serviços

A construção dos serviços, utilizando o *Webmethods Developer*, não necessitou na maior parte dos casos de um recurso a programação, bastando arrastar os vários componentes que a aplicação dispõem e conjuga-los de modo a criar um método semelhante ao que seria um método escrito por exemplo em Java.

Como é conhecido, uma das maiores vantagens de *SOA* é a diminuição de custos de desenvolvimento, pois permite a reutilização de funcionalidades já implementadas (permitindo também uma maior facilidade de integração entre funcionalidades de sistemas diferentes), e uma maior facilidade na manutenção das funcionalidades já existentes. Sendo assim, durante a construção de qualquer serviço fui sempre tendo em conta os seguintes aspectos [13]:

- **Atomicidade** – a implementação de um serviço deve ser centralizada na sua função, pois quanto mais complexo for as suas funcionalidades, menos reutilizável um serviço pode ser;
- **Generalização** – a implementação deve ser efectuada tendo em consideração a sua capacidade de reutilização em diferentes casos (como por exemplo em outros serviços), devendo ser mais generalizado possível, mas tendo sempre em conta a sua função principal;
- **Simplicidade e clareza** – a lógica de um serviço deve ser o mais simples e claro possível, para facilitar a sua manutenção.

De entre as várias regras impostas pela equipa ou até mesmo pelo cliente para a construção de serviços (por exemplo os serviços estarem organizados dentro de pacotes com nomes que sugiram facilmente o que fazem) saliento o facto de que, ao contrário do que intuitivamente foi definido na especificação dos serviços (dizendo que cada serviço recebe vários inputs de vários tipos de dados), o que aconteceu na realidade foi que para uma maior facilidade de uso e manutenção de variáveis internamente, essas vários inputs foram colocados dentro de documentos, estando os parâmetros de entrada num documento de entrada, e os parâmetros de saída num documento de saída, tendo estes documentos nomes únicos baseados no nome do serviço a que pertence (como por

exemplo <nome do serviço>*InputVars* e <nome do serviço>*OutputVars*). Esta maneira de implementação evita a substituição não intencionada de valores de alguma variável durante a construção do serviço pois é comum existirem variáveis com o mesmo nome que ficam na *pipeline* durante a execução de um serviço. É coberta assim uma potencial fonte de erros já previamente conhecidos na plataforma *webMethods*.

Na maioria dos serviços desenvolvidos existe uma variável status como output. Essa variável é preenchida com um de dois valores: “ok” e “fail”. Nas situações em que o status é devolvido com “ok” é indicado que o serviço correu todos passos correctamente, caso contrário é devolvido “fail”. Este status é particularmente importante dado que despoleta os *popups* de confirmação das acções que o utilizador executa no *Help Online*.

Relativamente à variável *size*, é comum encontrá-la em serviços que possuem no output listas. O *size* tal como o status permite verificar se tudo correu como pretendido dado que determinados serviços deverão sempre devolver listas preenchidas com um determinado número de valores. Caso o valor do *size* seja incoerente este poderá despoletar uma mensagem de erro tal como o status efectua.

Total de Serviços desenvolvidos: 34

Adapter Services

Nesta mesma camada, e usando a mesma ferramenta, foram também construídos vários *Adapter Services*. Como já foi explicado no contexto tecnológico, o *Integration Server* comunica com a camada da Base de Dados através de *JDBC adapters*. Os *adapters* que foram criados podem ter um código *SQL* (de selecção, actualização, inserção ou eliminação) relacionado, ou pode invocar um procedimento implementado na camada da Base de Dados. Estes *adapters* são posteriormente utilizados dentro dos serviços, tendo por norma um nome semelhante.

Total de Adapter Services desenvolvidos: 25

Transacções

Já com os *Serviços* e *Adapter Services* criados, foi necessário ter em atenção o uso de transacções, existindo já serviços próprios para início e fecho de transacções. Com uma análise prévia foram detectados possíveis problemas ao se verificar o uso de muitos *adapters* de escrita em quase todos os serviços. Sendo assim, durante a execução de uma transacção de escrita era possível a abertura de outro *adapter* que escrevesse no mesmo sítio sem que a 1ª tivesse terminado. Assim, enquanto uma transacção aberta não estiver fechada, o acesso a mesma conexão fica bloqueada. Foram assim respeitadas as regras **ACID** [9] que definem uma transacção. Passo a citar:

- **Atomicidade:** Após a transacção terminar (*commit* ou *abort*), a base de dados não deve reflectir resultados parciais da transacção (Faz tudo, ou não faz nada).
- **Consistência:** Regras de integridade da base de dados são sempre asseguradas.
- **Isolamento:** O trabalho é feito isoladamente. Operações externas a uma dada transacção não irão ver essa transacção em estados intermediários.
- **Durabilidade:** Os efeitos de uma transacção em caso de sucesso (*commit*) são permanentes mesmo em presença de falhas.

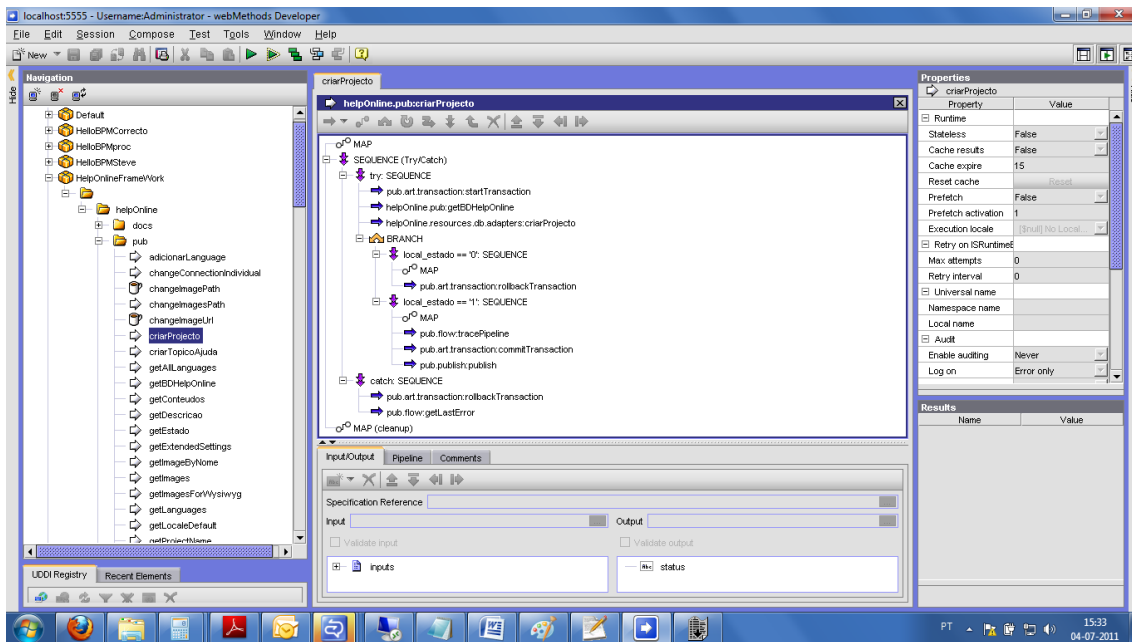


Fig. 22: Exemplo do Serviço 'inserirTemplDespacho' no Developer

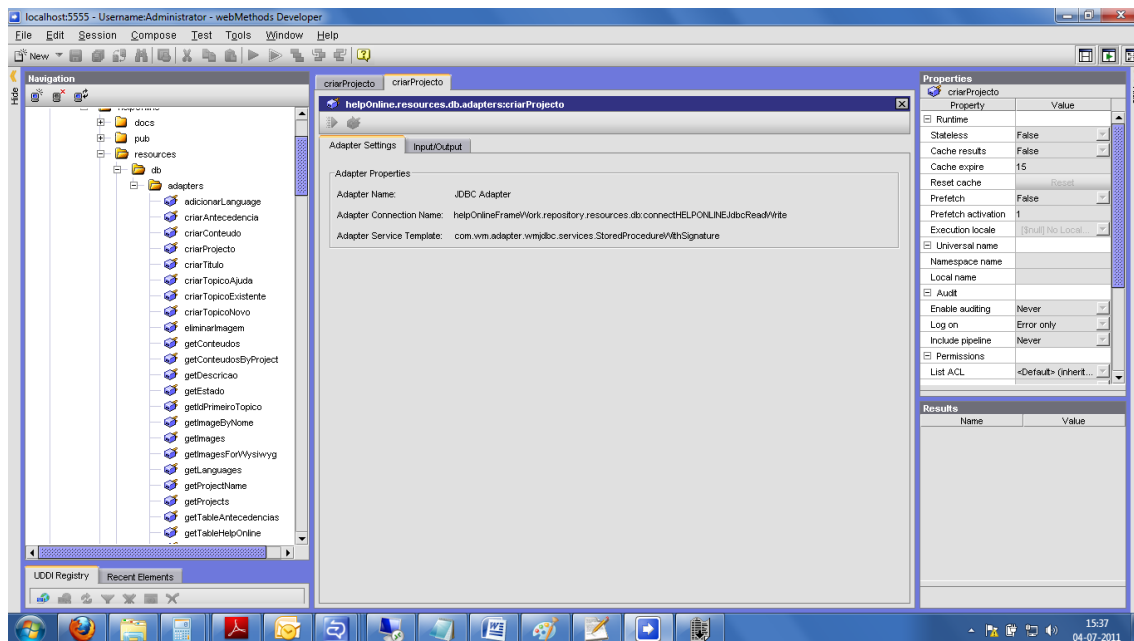


Fig. 23: Exemplo de um *Adapter Service* para Criar um Projecto

Principais linhas orientadores seguidas [11]:

- Convenção para nomes:
 - Os nomes dos serviços, *folders*, variáveis locais aos serviços, bem como as de input e output deverão ser escritos usando *camelCase* com a primeira letra minúscula;
- Boas práticas de desenvolvimento:
 - Usar ao máximo variáveis locais;
 - Minimizar o uso de *loops*;
 - Iniciar um documento *localvars* no início do serviço;
 - Limpar todas as variáveis não necessárias ao output do serviço no último *step*;
 - Colocar a inserção de dados dentro de um *try-catch*;
 - Comentar ao máximo cada serviço para facilitar a vida de quem tem de fazer manutenção;
 - Não deixar serviços em *lock 'ad eternum'*
 - Sempre que possível permitir que os serviços possuam um documento de status no output, de modo a saber se correu bem ou mal;

3.5.3 Camada de Dados

A camada de dados do *Help Online* representa os modelos de dados mostrados anteriormente e os mecanismos de acesso aos mesmos. O acesso a dados foi sempre realizado através da execução de *stored procedures* ao invés de um acesso mais directo. Estas *procedures* são necessários para a interacção entre o IS e os objectos da base de dados e respectivos dados, pois a maioria dos *Adapter Services* criados referenciavam estes procedimentos.

Nesta camada foram aplicadas várias normativas para nomenclatura dos objectos, que mais uma vez teriam que respeitar os nomes dos *Adapter Services* e dos anteriores serviços que o iriam chamar.

Por fim, é de referir que teve que ser criado um script de *export* da base de dados para passagem a produção. Esse script foi dividido:

- **Script com DDL**[10] (*Data Definition Language*): Script que permitiu definir as tabelas do Help Online e os elementos associados;
- **Script com DML**[10] (*Data Manipulation Language*): Script maioritariamente constituído com *inserts* necessários para o bom funcionamento da base de dados;

Total de Packages: 1 (PKG_HELP_ONLINE)

Total de Procedures: 25

```
create or replace PACKAGE          "PKG_HELP_ONLINE" AS
PROCEDURE  PRC_SET_ESTADO(ID_TOPICO  IN  varchar2,LOC  IN
varchar2,ESTADO IN varchar2);
END PKG_HELP_ONLINE;
```

Tabela 14: Stored Procedure Specification (sps)

```

create or replace PACKAGE BODY          "PKG_HELP_ONLINE" AS
PROCEDURE   PRC_SET_ESTADO(ID_TOPICO   IN   varchar2,LOC   IN
varchar2,ESTADO IN varchar2)
IS
id_conteudo varchar2(100);
locale_conteudo varchar2(100);
BEGIN
      select      topico_ajuda_conteudo_id,topico_ajuda_conteudo_locale      into
id_conteudo, locale_conteudo from topicos_ajuda
      where id = to_number(id_topico)
      and locale = LOC;

      UPDATE topico_ajuda_conteudo
      SET disabled = estado
      WHERE id = id_conteudo and locale = locale_conteudo;
END PRC_SET_ESTADO;
END PKG_HELP_ONLINE;

```

Tabela 15: Stored Procedure Body (spb)

Principais linhas orientadoras seguidas [11]:

- Oracle *naming conventions* [14];
- Não devem ser criados objectos fora de *packages* (procedimentos ou funções);
- Todo o código que funcionalmente esteja relacionado deve ser agrupado num *package*;
- Deve ser assegurado um nível de normalização adequado;
- Todo o código desenvolvido deve ser testado em termos de performance

3.5.4 Testes

Passando então para a fase de testes do sistema importa referir que o principal objectivo desta etapa passou por assegurar a qualidade do produto e um nível de confiança na utilização de forma a satisfazer os requisitos funcionais e não funcionais.

Para tal acontecer, foi necessário:

- Não deixar passar em claro possíveis defeitos na fase de construção do sistema;
- Assegurar que o produto respeita os requisitos funcionais e não funcionais atribuídos inicialmente.

Foram utilizadas as seguintes técnicas de verificação de qualidade:

- *Peer Reviews*[31]: Em parceria com outro colega, esta foi a forma mais rápida e fácil, e que por consequência consumiu menos tempo, para fazer uma análise a partes do código Java assim como grande parte do código feito na camada de dados. Como estavam os dois enquadrados na mesma tecnologia e a realizar tarefas semelhantes, cada um identificou erros do outro.
- Testes unitários para cada serviço, verificando se está de acordo com a sua especificação de input e output. Nesta fase o campo status falado anteriormente foi-nos bastante útil;
- Testes unitários a cada componente das interfaces. Foi feito segundo uma bateria de testes concebida pela equipa funcional, pois possuem uma lista pré concebida para problemas que acontecem a maior parte das vezes no desenvolvimento em *webMethods*.

Adicionalmente foi criado um diagrama (Fig. 24) para definir as relações existentes entre os requisitos identificados inicialmente e as capacidades do produto desenvolvido, assim como a prioridade tida em conta no seu cumprimento. O seu objectivo foi facilitar o *tracing* desde os requisitos até a um nível mais baixo, que neste caso diz respeito ao produto final desenvolvido.

A '*Relação Requisito-Interface*' representa a importância que o requisito tem na interface:

- Forte: o requisito é visível;
- Média: o requisito é visível indirectamente;
- Fraca: o requisito é suposto;

A 'Correlação interfaces' pretende demonstrar como as interfaces se afectam entre si:

- '++' onde o funcionamento de uma depende directamente da outra
- '+' onde o funcionamento de uma depende da outra mas a dependência não é directa
- '-' onde o funcionamento de uma afecta o funcionamento da outra

		Capacidades do Sistema Help Online				Aplicação do cliente	Prioridade (0-10)
		BackOffice		FrontOffice			
		Interface de Administração	Gestão do Help Online	Edição dos Tópicos de Ajuda	Interface de Utilizador	Interface a receber Help Online	
R e q u i s i t o s	HO_001				○		5
	HO_002	⊙	⊙	△	○		9
	HO_003					⊙	7
	HO_004			○			3
	HO_005	○					5
	HO_006	○	○	○	○		9
	HO_007			⊙			7
	HO_008			⊙			2
	HO_009	⊙					5
	HO_010	⊙					9
	HO_011	⊙					9
	HO_012		○				4
	HO_013		○				4
	HO_014		○	⊙			5
	HO_015					⊙	5
	HO_016	○	○		○		3
	HO_017	⊙					5
	HO_018	⊙					6
	HO_019		○				6
	HO_020		○				4
	HO_021				⊙	△	6
	HO_022	○					3
	HO_023		○				5
	HO_024	⊙					9
	HO_025			○			6
	HO_026		○				6
	HO_027	○					7

Fig. 24: Diagrama Requisitos vs Produto

3.6 Implementação e aceitação do sistema (IAS)

Por fim, esta fase visou a realização das actividades necessárias para a passagem à produção do sistema, onde de seguida são mostrados os resultados tangíveis.

O Sistema desenvolvido e que ficou a funcionar é constituído por 4 interfaces (Fig. 19) que podem por sua vez ser acedidas por 2 tipos de utilizadores, um com perfil de administração (BackOffice) e outro com perfil de utilizador genérico (FrontOffice), o qual dá uso às funcionalidades de ajuda on-line.

Toda a lógica desta aplicação é suportada pela criação de uma estrutura de dados em árvore onde são guardados os tópicos de ajuda. Deste modo foi-nos possível representar a sequência perfeita da navegação do portal do projecto, não esquecendo que os tópicos de ajuda possuem sempre um tópico antecedente e precedente (excepto o primeiro e os últimos) que possibilita a criação de um 'path' que por sua vez representa o caminho que esse utilizador percorreu.

A solução desenvolvida consistiu então no seguinte:

Interface de Administração (BackOffice)



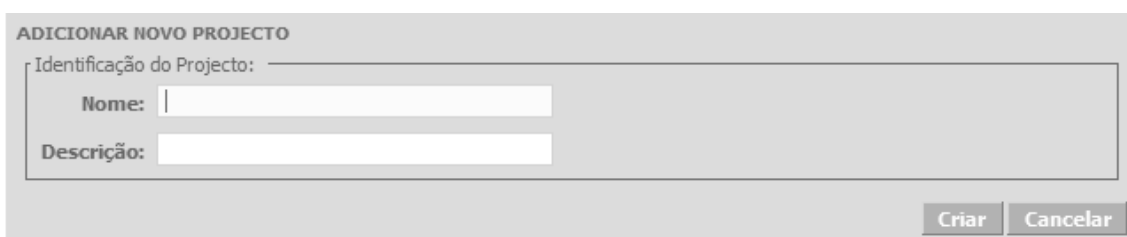
Fig. 25: Help Online Demo – Interface de Administração

O Ecrã de Administração do *Help Online* é acedido pela equipa de desenvolvimento do projecto onde se irá inserir o *Help Online*. Através desta interface de administração poderá criar/associar o(s) seu(s) novo(s) projecto(s) à aplicação Help Online e posteriormente adicionar novos tópicos de ajuda. Neste ecrã apenas é relevante criar a estrutura (em árvore) do projecto e dos conteúdos de ajuda. Os textos

explicativos e as imagens ilustrativas do tópico de ajuda ficarão a cargo do Ecrã Editar Tópico de Ajuda.

A utilização da interface de administração pressupõe que o primeiro passo seja a criação de um projecto. Esta acção irá efectuar inicializações na base de dados que permitirão posteriormente criar e gerir os tópicos de ajuda e imagens associados a um determinado projecto.

O primeiro componente do Ecrã de Administração do *Help Online* é portanto a Gestão de Projectos. O link [Adicionar Novo Projecto](#) irá despoletar o seguinte *popup*:

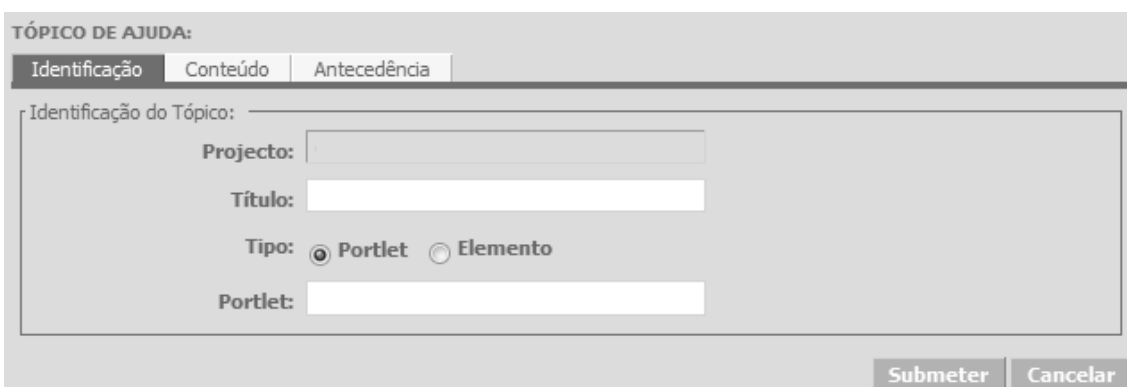


The screenshot shows a web form titled "ADICIONAR NOVO PROJECTO". It features a header with the title and a sub-header "Identificação do Projecto:". Below this, there are two input fields: "Nome:" and "Descrição:". At the bottom right of the form, there are two buttons: "Criar" and "Cancelar".

Fig. 26: Help Online Demo – *Popup* de Adicionar Novo Projecto

Após a criação do projecto, na *dropdown* Projecto passará a ser possível seleccionar o novo projecto criado aparecendo de seguida o segundo componente deste ecrã, a Gestão de Tópicos do Projecto.

Este componente apresenta uma árvore de tópicos de ajuda que inicialmente se encontra vazia. Será agora necessário inserir tópicos de forma a garantir que a estrutura criada corresponde a uma sequência perfeita da navegação do portal do projecto. É possível criar um novo, que despoletará o seguinte *popup*:



The screenshot shows a web form titled "TÓPICO DE AJUDA:". It has three tabs: "Identificação", "Conteúdo", and "Antecedência", with "Identificação" selected. Below the tabs, there is a sub-header "Identificação do Tópico:". The form contains several input fields: "Projecto:", "Título:", and "Portlet:". There are also two radio buttons for "Tipo:", with "Portlet" selected and "Elemento" unselected. At the bottom right, there are two buttons: "Submeter" and "Cancelar".

Fig. 27: Help Online Demo – *Popup* de Criação de Tópicos de ajuda (tab 1)

TÓPICO DE AJUDA:

Identificação | Conteúdo | Antecedência

Conteúdo:

Conteúdo: Criar Novo Associar Existente

Título do Novo Conteúdo:

Submeter Cancelar

Fig. 28: Help Online Demo – *Popup* de Criação de Tópicos de ajuda (tab 2)

TÓPICO DE AJUDA:

Identificação | Conteúdo | Antecedência

Tópico Antecedente:

Tópico Antecedente Seleccionado:

Projecto (pt)

LOGIN

AS MINHAS FARMÁCIAS

LISTA DE FARMÁCIAS

Submeter Cancelar

Fig. 29: Help Online Demo – *Popup* de Criação de Tópicos de ajuda (tab 3)

Neste *popup* o técnico tem disponíveis três separadores, um primeiro onde colocará os dados que identificarão tópico, um segundo onde indicará se pretende criar um novo conteúdo ou associar o conteúdo de um tópico já existente e por fim um terceiro onde é possível indicar em que ramo da árvore de conteúdos irá inserir-se o tópico criado.

Relativamente ao primeiro separador, Identificação, o técnico terá de fornecer o título do tópico, que identificará o tópico na árvore do projecto, uma informação sobre a *portlet* em que se insere e se for caso disso informação a que elemento da *portlet* se refere.

No separador Conteúdo existem duas hipóteses, criar um novo conteúdo, bastando para isso atribuir uma designação, ou então associar a este tópico de ajuda um conteúdo já existente, ou seja, um conteúdo que poderá já pertencer a um ou mais tópicos deste projecto.

Por fim o separador de Antecedência disponibiliza a árvore de tópicos do projecto. Se o técnico pretender adicionar o tópico que está a criar no nível mais baixo tem duas soluções, ou clicar no nome do projecto, no exemplo da imagem acima é PROJECTO

(PT) ou então simplesmente não definir nenhum tópico seleccionado. Caso pretenda adicionar noutro nível bastará clicar no tópico que quer passar a ter como antecedente.

Após correcta criação do tópico de ajuda é possível editá-lo. Para isso o técnico terá de seleccionar o tópico que pretende editar clicando no link correspondente na árvore de conteúdos e de seguida pressionar [Editar Tópico de Ajuda](#).

É despoletado um *popup* igual ao que aparece quando se clica em [Criar Novo Tópico de Ajuda](#) no entanto os campos vêm totalmente preenchidos com os dados do tópico. As imagens abaixo demonstrarão um exemplo do *popup* de edição:

The screenshot shows the 'Identificação' tab of a help topic editing popup. The title is 'TÓPICO DE AJUDA: AS MINHAS FARMÁCIAS'. The 'Identificação do Tópico' section contains the following fields: 'Projecto' (empty), 'Título' (filled with 'AS MINHAS FARMÁCIAS'), 'Tipo' (radio buttons for 'Portlet' (selected) and 'Elemento'), and 'Portlet' (empty). At the bottom right are 'Submeter' and 'Cancelar' buttons.

Fig. 30: Help Online Demo – *Popup* de Edição de Tópicos de ajuda (tab 1)

The screenshot shows the 'Conteúdo' tab of the same help topic editing popup. The 'Conteúdo' section has radio buttons for 'Criar Novo' and 'Associar Existente' (selected). The 'Conteúdo Seleccionado' field is filled with 'Conteúdo das minhas farmácias'. Below is a list of available content items under the heading 'CONTEÚDOS': 'Conteúdo de Login', 'Conteúdo da lista de farmácias', and 'Conteúdo das minhas farmácias'. A pagination bar shows '1 - 4 of 4'. At the bottom right are 'Submeter' and 'Cancelar' buttons.

Fig. 31: Help Online Demo – *Popup* de Edição de Tópicos de ajuda (tab 2)

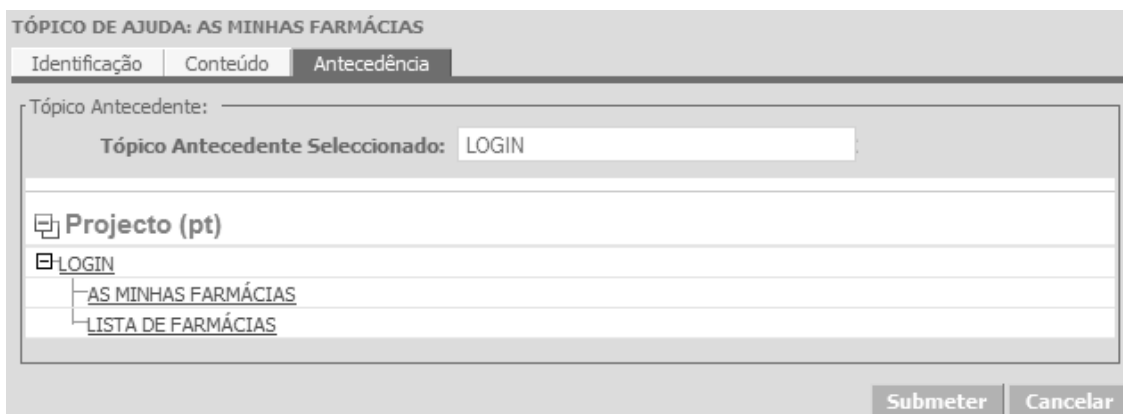


Fig. 32: Help Online Demo – *Popup* de Edição de Tópicos de ajuda (tab 3)

Os tópicos de ajuda de um projecto podem ser removidos. Para efectuar tal acção é necessário seleccionar o tópico que se pretende remover e de seguida utilizar o [link Remover Tópico de Ajuda](#).

De forma a facilitar a associação entre os [links Ajuda](#) existentes nos portais dos projectos que utilizam o *Help Online* e a própria estrutura de tópicos criada no Ecrã de Administração do *Help Online* adicionamos a opção de geração de um relatório com toda a informação identificativa dos tópicos de ajuda. A criação desse relatório é efectuada através do [link Gerar Relatório](#) que criará um documento em formato pdf onde constarão os *IDS*, títulos e conteúdos de cada tópico de ajuda de um determinado projecto. A imagem seguinte exemplifica um relatório de um projecto:

HELP ONLINE - Detalhes do Projecto

Projecto:		Gerado:	
Id:	62	Por:	Administrator
Nome:	Projecto	Em:	2011/01/14 10:44:39

Id:	Título:	Conteúdo Associado:
222	LOGIN	164
246	ALTERAR PASSWORD	190
223	AS MINHAS FARMÁCIAS	176
245	DADOS PESSOAIS	189
224	LISTA DE FARMÁCIAS	174
247	PESQUISA DE CONCURSOS	191
248	PESQUISA DE PREÇO	192

Fig. 33: Help Online Demo – Relatório de resumo do projecto

Interface de Gestão (BackOffice)

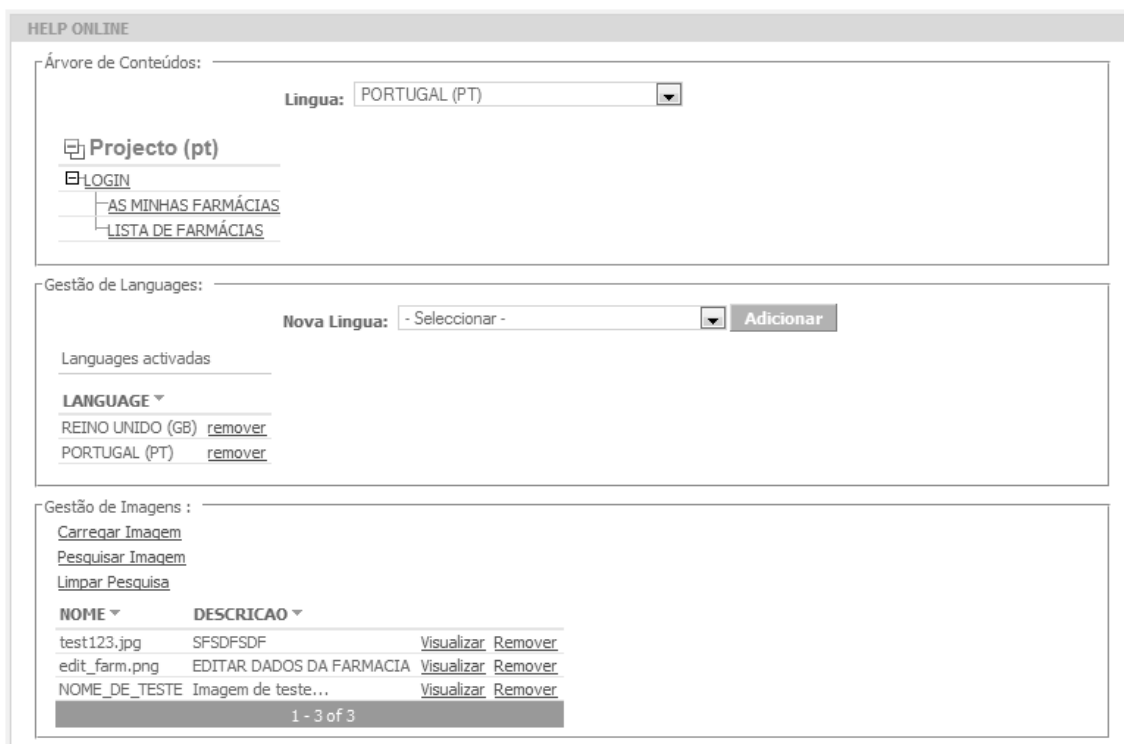


Fig. 34: Help Online Demo – Interface de Gestão

O Ecrã Gestão do *Help Online* é acedido através do BackOffice após um técnico clicar na opção *Help Online* no menu de uma determinada aplicação. Este acesso já foi explicado anteriormente neste documento no “Acesso ao *Help Online* – *BackOffice*”.

O Ecrã Gestão do *Help Online* está dividido em três componentes: Gestão de Conteúdos, Gestão de Línguas e Gestão de Imagens.

O primeiro componente do Ecrã Gestão de Tópicos de Ajuda corresponde a uma árvore de conteúdos onde constam todos os tópicos de ajuda da aplicação em que é possível editar o seu conteúdo. A estrutura desta árvore é gerida por um role de administração que utiliza uma interface exclusiva que permite adicionar projectos e editar as respectivas árvores de tópicos de ajuda (explicado anteriormente). A árvore criada para cada projecto deverá representar com fiabilidade o *Flow* de ecrãs da aplicação. Cada elemento da árvore poderá ser editado, bastando para isso clicar no tópico de ajuda que se pretende alterar. Automaticamente o utilizador é encaminhado para o Ecrã Editar Tópico de Ajuda que será posteriormente descrito detalhadamente.

O segundo componente deste ecrã permite efectuar a Gestão de Línguas associadas ao projecto, possibilitando ao técnico adicionar novas línguas aos projectos.

Na prática o técnico ao adicionar uma nova língua irá duplicar a árvore de tópicos existente mas para cada título do tópico de ajuda será adicionado a abreviatura da língua além do seu conteúdo ser totalmente limpo. Por exemplo, se um projecto tiver a seguinte árvore de tópicos:



Ao adicionarmos uma nova língua ao projecto, irá ser criada a seguinte árvore:



Ao clicarmos nos novos conteúdos FR estes estarão completamente limpos possibilitando ao técnico escrever o conteúdo mas agora na língua francesa. A alteração entre árvores é possível através da *dropdown* Língua existente na Gestão de Conteúdos deste ecrã e que possibilita a escolha entre uma das línguas previamente adicionadas.

O terceiro componente deste ecrã corresponde à Gestão de Imagens. Nesta área o técnico poderá carregar novas imagens para serem usadas em tópicos de ajuda, eliminar imagens já existentes e efectuar pesquisas por imagens já inseridas.

Para carregar uma nova imagem o técnico terá de clicar no *link* [Carregar Imagem](#) que despoletará o seguinte *popup*:

CARREGAR IMAGEM

Caminho:

Descrição:

Fig. 35: Help Online Demo – Popup de carregamento de Imagens

Neste *popup* o utilizador poderá indicar o local onde se encontra a imagem que pretende carregar e atribuir-lhe uma descrição que permitirá identifica-la mais facilmente. Após o técnico clicar no botão é adicionada uma referência na

base de dados e a imagem é carregada para o servidor passando a estar disponível para ser adicionada no editor de texto do Ecrã Editar Tópico de Ajuda.

Cada linha da tabela que lista as imagens inerentes ao projecto tem duas opções: [Visualizar](#) e [Remover](#).

Se o técnico clicar em [Visualizar](#) é despoletada uma nova janela contendo a imagem que o técnico pretende pré-visualizar.

Esta nova janela permite ao técnico visualizar a imagem facilitando a sua identificação.

Uma funcionalidade que poderá ser prática em projectos extensos com um grande número de imagens associadas é a pesquisa de imagens. O técnico se pretender poderá clicar em [Pesquisar Imagem](#) sendo despoletado o seguinte *popup*:

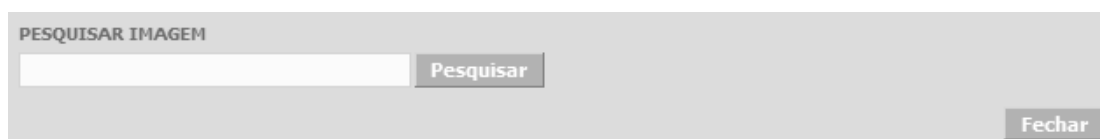
A screenshot of a web application's search popup. The popup has a light gray background and a title bar at the top that reads "PESQUISAR IMAGEM". Below the title bar, there is a white text input field. To the right of the input field is a gray button with the text "Pesquisar". In the bottom right corner of the popup, there is another gray button with the text "Fechar".

Fig. 36: Help Online Demo – *Popup* de Pesquisa de Imagem

Neste *popup* o técnico poderá inserir o critério de pesquisa que pretende e posteriormente a clicar no botão [Pesquisar](#). A tabela de imagens é actualizada demonstrando apenas as imagens cujo nome do ficheiro vai de encontro ao colocado no critério de pesquisa. Para a tabela voltar a listar a totalidade das imagens inerentes ao projecto bastará clicar em [Limpar Pesquisa](#) ou então voltar a despoletar o *popup* de pesquisa de imagens e efectuar uma pesquisa sem critério.

Interface de Gestão – Edição dos Tópicos de Ajuda (BackOffice)

EDITAR TÓPICO DE AJUDA

Informação:

TIPO: ANTECEDENTES ▾
Portlet: LOGIN

Conteúdo:
 activo desactivo

Título: LOGIN

Terminar Cancelar

Fig. 37: Help Online Demo – Ecrã Editar Tópico de Ajuda

ESCOLHA DE IMAGEM

Manual Pesquisar

GestãoTópicosAjuda.png ▾

Imagem Seleccionada [Visualizar:](#) _____

Nome: GestãoTópicosAjuda.png

Descrição: Imagem para gestão de tópicos de ajuda

Inserir Cancelar

Fig. 38: Help Online Demo – *Popup* Escolha Manual de Imagem

ESCOLHA DE IMAGEM

Manual Pesquisar

GestãoTópicos | Pesquisar

Selecione a Imagem: _____

Imagens	
NOME ▾	DESCRICAO ▾
GestãoTópicosAjuda.png	Imagem para gestão de tópicos de ajuda

1 - 1 of 1

Imagem Seleccionada [Visualizar:](#) _____

Nome: GestãoTópicosAjuda.png

Descrição: Imagem para gestão de tópicos de ajuda

Inserir Cancelar

Fig. 39: Help Online Demo – *Popup* Escolha De Imagem por Pesquisa

O Ecrã Editar Tópico de Ajuda permite que um técnico com perfil de gestão do conteúdo do *Help Online* edite os tópicos de ajuda existentes na árvore de conteúdos do Ecrã Gestão do *Help Online*.

Este ecrã é despoletado após o técnico no Ecrã de Gestão do *Help Online* escolher na árvore de conteúdos o tópico que pretende editar.

O Ecrã Editar Tópico de Ajuda é composto por dois grupos, sendo um dedicado a informação sobre o tópico em questão, e sendo outro dedicado à edição do seu conteúdo. A secção de informação disponibiliza ao utilizador o formato do tópico de ajuda (que pode ser uma *portlet*, ou apenas uma componente/elemento da *portlet*) assim como a sua localização física (antecedências) na estrutura da aplicação. Na secção de conteúdo são presentes os campos “título”, “activo/desactivo” e um editor de texto que designamos por “WYSIWYG”. O campo título não é mais que uma mera designação dada ao tópico de ajuda, sendo que esta designação é a que consta na árvore de conteúdos do Ecrã Gestão do *Help Online*. O campo “activo/desactivo” serve para activar ou desactivar o conteúdo de um determinado tópico de ajuda, possibilitando a ocultação do seu conteúdo aquando de uma consulta na árvore de consulta. Finalmente o “WYSIWYG” é uma ferramenta de edição de texto que possibilitará ao técnico editar o conteúdo de ajuda que pretende disponibilizar aos utilizadores do portal, tendo a particularidade de disponibilizar diversas opções intuitivas tal como inserção de imagens e formatação de texto. De referir que a opção de inserção de imagens disponibilizada pela WYSIWYG foi melhorada, disponibilizando uma consulta à base de dados (através de uma *dropdown*) que devolve todas as imagens já carregadas anteriormente para um determinado projecto, bastando para isso estar seleccionando a opção Manual . É possível também efectuar uma pesquisa (clicando na opção Pesquisar) de uma determinada imagem, caso a lista seja extensa. Para visualizar a imagem pretendida terá que se clicar em [Visualizar](#).

Interface de Utilizador (*FrontOffice*)

HELP ONLINE

Menu de pesquisa: _____

Tópico de ajuda: _____

CONTEUDO DO TOPICO DE AJUDA

Fig. 40: Help Online Demo – Ecrã de Consulta de Tópicos de Ajuda (1)

HELP ONLINE

Menu de pesquisa: _____

Pesquisa Manual | Pesquisa por Tópico

- PROJECTO
- LOGIN
 - AS MINHAS FARMÁCIAS
 - LISTA DE FARMÁCIAS

Tópico de ajuda: _____

CONTEUDO DO TOPICO DE AJUDA

Fig. 41: Help Online Demo – Ecrã de Consulta de Tópicos de Ajuda (2)

HELP ONLINE

Menu de pesquisa: _____

Pesquisa Manual | Pesquisa por Tópico

Critério de Pesquisa:

LOGIN -> AS MINHAS FARMÁCIAS
LOGIN -> LISTA DE FARMÁCIAS

Tópico de ajuda: _____

CONTEUDO DO TOPICO DE AJUDA

Fig. 41: Help Online Demo – Ecrã de Consulta de Tópicos de Ajuda (3)

HELP ONLINE

Menu de pesquisa: _____

Tópico de ajuda: _____

DE MOMENTO NÃO EXISTE AJUDA DISPONIVEL. PODERÁ UTILIZAR O MENU DE PESQUISA.

Fig. 42: Help Online Demo – Ecrã de Consulta de Tópicos de Ajuda (4)

O Ecrã de Consulta de Tópicos de Ajuda é acedido quando o utilizador clica na hiperligação [Ajuda](#) presente no portal da aplicação para o qual se fez o *Help Online*, sendo despoletada uma nova janela onde aparece a aplicação *Help Online*.

O primeiro componente do ecrã é uma secção onde constam as operações de pesquisa de tópicos de ajuda. A primeira opção é chamada de “Pesquisa Manual” e permite ao utilizador consultar a estrutura base de como está organizado o *Flow* de ecrãs do portal da aplicação (entenda-se *Flow* como a sequência de ecrãs, dependentes entre si). Poderá a qualquer momento clicar no tópico de ajuda desejado, abrindo na secção em baixo o respectivo conteúdo do tópico. A segunda opção é chamada de “Pesquisa por Tópico” e permite ao utilizador fazer uma pesquisa sobre algum assunto desejado. Dessa forma, insere o critério de pesquisa sendo devolvidos os caminhos existentes no *Flow* de ecrãs que terminem na palavra ou expressão dada pelo utilizador, podendo o utilizador utilizar os *links* devolvidos para abrir por baixo o conteúdo desse tópico de ajuda.

O segundo componente do ecrã é a secção onde é aberto o conteúdo de um tópico de ajuda, quer pelo facto utilizador clicar num dos *links* devolvidos pela “primeira componente”, quer por o utilizador clicar em [Ajuda](#) durante a navegação na aplicação. Caso o tópico que pretende consultar esteja desactivado, irá aparecer a seguinte mensagem: "De momento não existe ajuda disponível. Poderá utilizar o menu de pesquisa”.

Capítulo 4

Conclusão

O estágio foi realizado no âmbito da cadeira Projecto de Engenharia Informática do Mestrado em Engenharia Informática área de especialização Sistemas de Informação da Faculdade de Ciências da Universidade de Lisboa e deu origem a um sistema de ajuda contextualizada no qual se consegue editar online os conteúdos de ajuda para determinada interface ou componente dela.

O trabalho iniciou-se com integração na empresa e posteriormente procedeu-se à integração no cliente/projecto. De seguida realizou-se a análise de documentação existente de modo a compreender o negócio assim como a tecnologia a ser usada.

A solução foi então desenvolvida na plataforma *webMethods*, utilizando a arquitectura SOA, tendo o *Help Online* permitido, de forma lata, o seguinte:

- Disponibilizar ajuda para cada ecrã da aplicação ou componente do mesmo, podendo esta ajuda ser editada online e sendo fornecido ao utilizador uma ferramenta de pesquisa de ajuda;
- Criar um sistema de Ajuda para a maioria das aplicações desenvolvidas em *webMethods*;

As vantagens deste projecto residiram essencialmente nos ganhos de tempo por parte do utilizador na procura de uma solução para o seu problema pontual na aplicação e também nos ganhos para a empresa, visto o sistema ser adaptável a todos as aplicações em *webMethods*.

Durante todo o estágio o aluno desenvolveu várias capacidades na vertente técnica, funcional e até mesmo na gestão de actividades. Tecnicamente consegui grande evolução na utilização do *webMethods*, apesar de saber inicialmente que a curva de aprendizagem seria bastante longa visto ser uma tecnologia difícil, onde praticamente não existe documentação. Visto estar presente em todo o processo de desenvolvimento

do sistema, foi possível rever e aplicar muitos dos conhecimentos adquiridos na faculdade, conhecimentos a que até agora não dava a devida importância. A nível de gestão, visto existirem prazos a cumprir, foi também possível evoluir na capacidade de organização de projectos de desenvolvimento de software.

Grande parte da positiva adaptação se deveu á boa formação académica prestada pela faculdade, nomeadamente nas disciplinas que envolviam programação em Java, como também nas disciplinas menos técnicas como Análise e desenho de Sistemas de Informação. A cadeira de aplicações na Web teve um papel muito importante pois proporcionou o contacto directo com a arquitectura SOA e com Javascript, o que facilitou a compreensão do funcionamento de grande parte da plataforma.

Em suma, apesar de algumas dificuldades provenientes principalmente da complexidade da tecnologia usada, o saldo foi bastante positivo e todos os objectivos iniciais foram cumpridos, salientando apenas algumas pequenas mudanças no planeamento mas que em nada afectaram o sucesso do projecto.

Capítulo 5

Bibliografia

- [1] <http://www.baremetalsoft.com/baretail/> - 15/08/2011
- [2] <http://tortoisesvn.net/> - 15/08/2011
- [3] <http://www.oracle.com/technetwork/developertools/sqldeveloper/overview/index.html> - 15/07/2011
- [4] <Http://www.slideshare.net/garrettdimon/improving-interface-design> - 15/07/2011
- [5] Sun Microsystems, Inc, Java Look And Feel Design Patterns, 1999 - 15/07/2011
- [6] http://pt.wikipedia.org/wiki/Modelo_relacional - 10/08/2011
- [7] http://pt.wikipedia.org/wiki/Modelo_em_tr%C3%AAs_camadas - 10/08/2011
- [8] <http://faculty.washington.edu/jtenenbg/courses/360/f04/sessions/schneidermanGoldmanRules.html> - 31/08/2011 - 10/08/2011
- [9] <http://pt.wikipedia.org/wiki/ACID> - 10/08/2011
- [10] <http://pt.wikipedia.org/wiki/SQL> - 10/08/2011
- [11] Normas_Desenvolvimento_Cliente.docx (documentação interna) – 15/08/2011
- [12] <http://java.sun.com/docs/codeconv/> - 31/08/2011
- [13] http://en.wikipedia.org/wiki/Service-oriented_architecture - 10/08/2011
- [14] <http://ss64.com/ora/syntax-naming.html> - 10/08/2011
- [15] http://pt.wikipedia.org/wiki/Business_Process_Management - 10/08/2011
- [16] <http://pt.wikipedia.org/wiki/Portlet> - 31/08/2011
- [17] <http://pt.wikipedia.org/wiki/XQuery> - 31/08/2011
- [18] <http://pt.wikipedia.org/wiki/Middleware> - 31/08/2011
- [19] <http://pt.wikipedia.org/wiki/Jetty> - 31/08/2011
- [20] <http://pt.wikipedia.org/wiki/Camelcase> - 31/08/2011
- [21] <http://pt.wikipedia.org/wiki/WebServices> - 31/08/2011

- [22] http://communities.softwareag.com/ecosystem/communities/public/Developer/webmethods/tutorials/CAF/CAFandJSF/JSF_Overview.html#1098892 – 31/08/2011
- [23] http://en.wikipedia.org/wiki/JavaServer_Faces - 15/08/2011
- [24] <http://en.wikipedia.org/wiki/SOAP> - 15/08/2011
- [25] Understanding the webMethods Suite Version 7.1, Darmstadt, Germany: Software AG, 2008
- [26] Getting Started with Business Process Management Version 7.1, Darmstadt, Germany: Software AG, 2007.
- [27] webMethods Broker Administrator's Guide Version 7.1, Darmstadt, Germany: Software AG, 2007.
- [28] webMethods Process Engine User's Guide 7.1.1, Darmstadt, Germany: Software AG, 2007.
- [29] webMethods Task Engine User's Guide Version 7.1.1, Darmstadt, Germany: Software AG, 2007.
- [30] webMethods Designer - Process Development Help Version 7.1.1, Darmstadt, Germany: Software AG, 2008.
- [31] http://en.wikipedia.org/wiki/Software_peer_review - 15/08/2011

Capítulo 6

Anexos

Glossário

<i>Soft-skills</i>	As <i>soft-skills</i> representam as atitudes e comportamentos das pessoas em interacção com outras. São atitudes comportamentais por natureza e desenvolvem-se através da prática em situações de trabalho reais.
<i>Webmethods</i> (empresa)	Empresa especializada em ferramentas de integração, fundada por Phillip Merrick em 1997. Adquirida pela <i>Software AG</i> em 2007.
<i>Webservices</i>	Um <i>WebService</i> pode definir-se como um módulo de uma aplicação, acessível via <i>Web</i> e que fornece para o exterior uma funcionalidade ou serviço, de forma a poder ser usado por um cliente. [21]
<i>WebMethods Designer</i>	É uma ferramenta baseada no <i>Eclipse</i> e que pertence à suite de <i>webMethods</i> . Permite o desenvolvimento de componentes gráficas através da tecnologia CAF e a modelagem de processos BPM.
<i>WebMethods Developer</i>	Ferramenta de desenvolvimento pertencente à suite de <i>webMethods</i> , que permite o desenvolvimento de <i>webservices</i> através de <i>Flow</i> .
<i>Service Oriented Architecture (SOA)</i>	É uma arquitectura em que as aplicações desenvolvidas devem usar funcionalidades oferecidas por serviços de um sistema diferente, sendo que estes serviços poderão ser usados por várias aplicações. Os serviços implementados oferecem às aplicações uma interface, através por exemplo dos <i>webServices</i> , anteriormente falados.
<i>XML Query Language</i>	<i>XQuery</i> é uma linguagem de consulta com o objectivo de fazer consultas em conjuntos de dados escritos em XML. É semanticamente similar ao SQL. [17]
<i>Middleware</i>	O termo <i>Middleware</i> refere-se ao <i>software</i> que se executam entre as

	aplicações e os sistemas operativos. O objectivo do middleware é facilitar o desenvolvimento de aplicações, tipicamente as distribuídas. [18]
<i>Business Process Management (BPM)</i>	O BPM é um conceito que une a gestão de negócios à tecnologia da informação, com o objectivo de melhorar os resultados das organizações através da optimização dos processos do negócio. [15]
<i>Workflow</i>	Conjunto de tarefas que constituem um processo de negócio.
<i>Hypertext Transfer Protocol (HTTP)</i>	Protocolo de comunicação na camada de aplicação utilizado nos sistemas distribuídos.
<i>eXtensible Markup Language (XML)</i>	Linguagem que permite representar vários tipos de dados, facilitando a partilha dos mesmo entre aplicações desenvolvidas em linguagens diferentes e na internet.
<i>Eclipse</i>	IDE para o desenvolvimento de aplicações em <i>Java</i> ou outras linguagens.
<i>Portlet</i>	Componente visual independente que pode ser utilizado para disponibilizar informações dentro de uma página Web. [16]
<i>Composite Application Framework (CAF)</i>	Tecnologia que permite o desenvolvimento de aplicações compostas, que podem aceder a dados de diferentes sistemas. CAF pode ser considerada como uma extensão da tecnologia JSF, pois disponibiliza controlos JSF e controlos com funcionalidades mais complexas implementadas sobre controlos JSF.
<i>Java Server Faces (JSF)</i>	Estrutura que permite o desenvolvimento de aplicações Web de forma visual através de arrastamento de componentes visuais para uma área de edição. Permite também a implementação de ecrãs em XML, que depois podem ser interpretados e renderizados para HTML.
<i>Asynchronous Javascript and XML (Ajax)</i>	Representa o uso metodológico de tecnologias como <i>Javascript</i> e XML para tornar as páginas Web mais interactivas com o utilizador, valendo-se de solicitações assíncronas de informações.
<i>Jetty</i>	O <i>Jetty</i> é um servidor HTTP e <i>Servlet Container</i> 100% escrito em Java. A grande vantagem do <i>Jetty</i> é a sua fácil configuração [19].
<i>Integrated Development Environment (IDE)</i>	Ferramenta de desenvolvimento que permite o desenvolvimento de aplicações.
<i>Flow</i>	Linguagem de programação gráfica incorporada na ferramenta <i>webMethods Developer</i> , para desenvolver <i>webservices</i> .

<i>Java</i>	Linguagem de programação orientada a objectos desenvolvida pela Sun <i>Microsystems</i> .
<i>Simple Object Access Protocol (SOAP)</i>	Protocolo utilizado para a troca de informações nos sistemas distribuídos, recorrendo a outros protocolos como RPC e HTTP. O formato de mensagem de SOAP tem como base o XML, permitindo assim a circulação de dados entre sistemas numa linguagem normalizada.
<i>Remote Procedure Call (RPC)</i>	Permite efectuar uma chamada a um procedimento que se encontra implementado num outro sistema através do acesso por rede.
Sistema de Gestão de Base de Dados (SGBD)	Um Sistema de Gestão de Base de Dados é responsável pela gestão de uma base de dados. O seu principal objectivo é retirar da aplicação cliente a responsabilidade de gerir o acesso, manipulação e organização dos dados.
<i>Log</i>	Descreve o processo de registo de eventos relevantes numa dada aplicação.
<i>SubVersion (SVN)</i>	Sistema de controlo de versões.
<i>Secure Shell (SSH)</i>	O SSH é simultaneamente um programa e um protocolo de rede que permite a conexão com outro computador na rede, de forma a executar comandos de uma unidade remota.
Telnet	Telnet é um protocolo cliente-servidor usado para permitir a comunicação entre computadores ligados numa rede. Baseado em TCP.
<i>Drag-and-drop</i>	<i>Drag-and-drop</i> representa a acção de clicar num objecto virtual e "arrastá-lo" para uma posição diferente ou sobre um outro objecto virtual.
<i>Pipeline</i>	Memória temporária interna do servidor onde são guardados valores temporários criados durante a execução de serviços ou outras componentes.
<i>Java Database Connectivity (JDBC)</i>	Definição utilizada em Java acerca de como é que o acesso à base de dados é efectuado.
<i>Structured Query Language (SQL)</i>	Linguagem de pesquisa declarativa utilizada em bases de dados relacionais.
ACID	ACID é um conceito utilizado para caracterizar uma transacção.
<i>Commit</i>	Refere-se à ideia de fazer permanentes um conjunto de mudanças experimentais.

<i>Camelcase</i>	É a denominação em inglês para a prática de escrever palavras compostas ou frases, onde cada palavra é iniciada com Maiúsculas e unidas sem espaços. [20]
<i>Stored procedure</i>	Conjunto de comandos em SQL. Encapsula tarefas repetitivas, aceita parâmetros de entrada e retorna um valor de <i>status</i> .
<i>Popup</i>	O <i>popup</i> é uma janela extra que se abre no browser ao visitar uma página web ou aceder a uma hiperligação específica.

Requisitos Funcionais

Código Requisito	Descrição Requisito	Dependências
HO_001	O Help Online é uma aplicação gestora da informação de ajuda que é disponibilizada aos utilizadores externos das aplicações.	
HO_002	Toda a informação é organizada em tópicos de ajuda.	
HO_003	Os tópicos de ajuda criados poderão ser associados aos ecrãs ou a componentes desses ecrãs.	
HO_004	Os tópicos de ajuda deverão ter a possibilidade de ser editados por um técnico que apresente um perfil adequado.	
HO_005	Os tópicos de ajuda deverão ser criados e organizados pela equipa que desenvolveu o projecto a necessitar de Help Online.	
HO_006	O Help Online é composto por duas funcionalidades principais: a de consulta, que é acessível através do frontoffice do projecto em questão; a de edição, que é acessível através do backoffice do projecto em questão. Adicionalmente poderá ser criada a funcionalidade de Administração para facilitar inserções na base de dados	
HO_007	A informação textual e ilustrativa dos tópicos de ajuda que se querem editar deverá ser inserida recorrendo a um editor de texto intuitivo cujo ambiente se assemelhe aos mais comuns.	
HO_008	O utilizador a editar um tópico de ajuda apenas poderá modificar o seu título, estado e conteúdo.	
HO_009	O elemento da equipa de desenvolvimento terá que utilizar a interface de administração para construir a árvore de conteúdo, que posteriormente será editada pelo técnico.	
HO_010	A árvore de conteúdos terá que ser dinâmica, com possibilidade de ser reajustada a qualquer momento	
HO_011	Os tópicos de ajuda deverão estar associados entre si, estabelecendo uma árvore hierárquica que tenderá a corresponder ao fluxo de ecrãs da aplicação.	
HO_012	Deverá existir uma área de gestão de imagens, onde o gestor poderá carregar e eliminar os conteúdos ilustrativos que insere nos tópicos.	
HO_013	Cada imagem deverá conter um campo descrição.	
HO_014	Cada tópico de ajuda tem associados apenas um título e um conteúdo. Esses títulos e conteúdos por sua vez podem ser representados em várias línguas.	
HO_015	É necessário a criação de Links para aceder ao Help Online. Esses links serão colocados pela equipa que desenvolve o projecto, na interface	

Código Requisito	Descrição Requisito	Dependências
	do portal referente ao projecto.	
HO_016	A árvore de conteúdos é apresentada utilizando os títulos dados ao tópicos de ajuda.	
HO_017	A cada ecrã/elemento/componente da aplicação só pode estar associado um único tópico de ajuda.	
HO_018	Vários tópicos de ajuda podem ter o mesmo conteúdo de ajuda.	
HO_019	Um conteúdo poderá estar disponível em diversas línguas.	
HO_020	Uma árvore de conteúdos poderá ser copiada para uma árvore idêntica noutra língua.	
HO_021	Deverá ser possível desactivar um tópico de ajuda, sendo mostrada ao utilizador que tenta aceder a esse tópico uma mensagem a indicar que de momento não existe ajuda disponível mas que poderá utilizar as funcionalidades de pesquisa.	
HO_022	Os tópicos criados no Ecrã de Administração deverão, por “default”, estar com um estado “desactivo”.	
HO_023	Após o carregamento de uma imagem é possível efectuar uma pesquisa da mesma, introduzindo um critério de pesquisa.	
HO_024	Existe um Help Online por projecto.	
HO_025	A barra de ferramentas do editor de texto deverá possuir botões que permitam apenas efectuar funcionalidades básicas. Não deverão constar por esta razão botões de copiar, colar, inserção de tabelas, escolha do tipo de letra, visualização html...	
HO_026	Quem edita um tópico deverá conseguir ver qual a sua localização real na árvore de conteúdos.	
HO_027	A qualquer momento é possível gerar um relatório que descreve todas as informações de cada projecto para ser possível montar o seu HelpOnline (ex: associar parâmetros que são enviados em Links)	

Requisitos Não Funcionais

Código Requisito	Descrição Requisito	Dependências
HO_NF_001	A funcionalidade de Help Online deverá ser ampla tendo a possibilidade de estar presente em todos os ecrãs e componentes desejadas que os utilizadores externos consigam aceder.	
HO_NF_002	A edição dos conteúdos do Help Online deverá ser efectuada de forma rápida e intuitiva, minimizando eventuais dúvidas de processamento de texto e inserção/eliminação de conteúdo ilustrativo.	
HO_NF_003	O Help Online deverá elucidar o gestor que a árvore de conteúdos corresponde às sequências de ecrãs.	
HF_NF_004	As funcionalidades menos inteligíveis deverão estar associadas a estruturas e diagramas que as simplifiquem, como “toggle trees” e “breadcrumbs”.	
HF_NF_005	O Help Online deverá ser uma funcionalidade dinâmica, tendo capacidade de ser adaptada a distintas estruturas aplicacionais.	
HF_NF_006	A gestão de conteúdos do Help Online não deverá requerer qualquer tipo de desenvolvimento programático. O sistema de Help Online deverá por esta razão ser autónomo e o mais personalizável possível.	

Serviços

Neste ponto serão descritas as funções de cada serviço existente no package HelpOnlineFramework no servidor IS.

Para cada serviço irão ser indicados os inputs e os outputs e que acções efectuam ao nível da base de dados.

adicionarLanguage

Nome: adicionarLanguage

Inputs: Loc, user, default_loc, projecto.

Outputs: status.

Dependências: N/A.

Descrição: Adiciona uma nova língua ao projecto que é passado como input. Na prática replica a estrutura de tópicos de ajuda existente para a “default” language, no entanto os novos conteúdos criados estarão a null e os títulos serão preenchidos com os da “default” language seguidos do alpha2code da nova língua.

changeImagePath

Nome: changeImagePath

Inputs: id_projecto, novo_path.

Outputs: status.

Dependências: N/A.

Descrição: Para cada conteúdo inerente a um determinado projecto existente na tabela TOPICO_AJUDA_CONTEUDO actualiza os valores da tag HTML “” de forma a actualizar a localização das imagens para o novo “path” passado como input do serviço. É igualmente efectuada a actualização do campo URL na tabela IMAGENS_REFDATA referente a cada imagem associada ao projecto.

criarProjecto

Nome: criarProjecto

Inputs: nome, descricao, default_language, updated_by

Outputs: status.

Dependências: N/A.

Descrição: Cria a estrutura na base de dados necessária para começar a introduzir tópicos de ajuda.

criarTopicoAjuda

Nome: criarTopicoAjuda

Inputs: id_projecto, loc, tipo, portlet, elemento, updated_by, titulo, accao_conteudo, id_conteudo, descricao_conteudo, id_antecedente.

Outputs: status.

Dependências: N/A.

Descrição: Este serviço pode seguir dois caminhos independentes consoante o que recebe o input accao_conteudo. No caso do accao_conteudo vier preenchido com o valor “existente”, significa que se está a tentar criar um novo tópico de ajuda, no entanto, o conteúdo que se lhe irá atribuir é um conteúdo já existente na base de dados desta forma apenas será efectuado um insert na tabela TOPICOS_AJUDA. Para o caso em que o input accao_conteudo vem preenchido com o valor “novo” terão de ser executados dois inserts, um primeiro na tabela TOPICO_AJUDA_CONTEUDO e outro na tabela TOPICOS_AJUDA sendo passado o GUID dos TOPICO_AJUDA_CONTEUDO como chave estrangeira do TOPICOS_AJUDA. É efectuado também um insert na tabela ANTECEDENCIAS de forma a colocar o novo tópico de ajuda no ramo correcto da árvore.

getAllLanguages

Nome: getAllLanguages

Inputs: N/A.

Outputs: listaPaises (codigo, nome), status.

Dependências: N/A.

Descrição: Devolve uma lista com todos os países existentes em base de dados. Para cada país é indicado um código (abreviatura de duas letras) e o nome.

getBDHelpOnline

Nome: getBDHelpOnline

Inputs: N/A.

Outputs: BDHelpOnline (Documento canónico cujo conteúdo representa fielmente todos os dados existentes em base de dados na altura em que o serviço é corrido)[].

Dependências: adicionarLanguage, changeImagesPath, criarProjecto, criarTopicoAjuda, insereHistorico, insertImage, removeImage, removeTopico, removerLanguage, removerProjecto, updateTopico, updateTopicoAjuda.

Descrição: Preenche o documento canónico com todos os dados existentes na base de dados referentes ao Help Online. Este documento irá servir como input do serviço insertBDHistory que guarda no histórico a base de dados do Help Online.

getConteudos

Nome: getConteudos

Inputs: id_projecto, language.

Outputs: conteudos (GUID, rule_id, conteudo)[], status, size.

Dependências: N/A.

Descrição: Devolve uma lista com todos os conteúdos existentes na base de dados na tabela TOPICO_AJUDA_CONTEUDO inerentes a um determinado projecto e lingua. São indicados os campos GUID, RULE_ID (neste resultado representa um género de um titulo do conteúdo) e CONTEUDO (código HTML que irá representar o conteúdo a apresentar no portal quando um utilizador pretende visualizar o tópico de ajuda).

getDescricao

Nome: getDescricao

Inputs: nome, projecto.

Outputs: descricao, status.

Dependências: N/A.

Descrição: Serviço que dado o nome de uma imagem e o id do projecto em que se insere devolve a sua descrição. Ele acede à tabela IMAGENS_REFDATA na base de dados.

getEstado

Nome: getEstado

Inputs: id, loc, projecto.

Outputs: disabled, status.

Dependências: N/A.

Descrição: Serviço que vai á tabela TOPICO_AJUDA_CONTEUDO e verifica se um determinado conteúdos está activo ou desactivo. Se o output disabled vier preenchido a 0 então está desactivado, se vier preenchido a 1 então o conteúdo está activo.

getExtendedSettings

Nome: getEstado

Inputs: path1, path2.

Outputs: result1, result2.

Dependências: N/A.

Descrição: Serviço vai às extended settings do servidor IS e preenche os outputs com os valores obtidos pelas properties passadas pelos path1 e path2. Normalmente as properties utilizadas pelo projecto Help Online são: “watt.server.cliente.helponline.images1” e “watt.server.cliente.helponline.images2”.

getImageByName

Nome: getImageByName

Inputs: nome, projecto.

Outputs: results(GUID, NOME_FICHEIRO, DESCRICAO, URL, HELP_ONLINE_ID, DOC_ID), status

Dependências: N/A.

Descrição: Este serviço devolve informação relativa à imagem cujo nome do ficheiro corresponde ao dado como input. Devolve para cada imagem o GUID (identificação

unívoca), NOME_FICHEIRO (nome do ficheiro referente à imagem, por exemplo “teste.jpg”), DESCRICAO (campo que permite aos utilizadores identificarem mais facilmente a imagem, sendo preenchido por um titulo ou resumo da imagem), URL (localização da imagem no servidor, terá de ser passada para o módulo WYSIWYG através de um protocolo http), HELP_ONLINE_ID (chave estrangeira da tabela HELP_ONLINE, indicando a que projecto está associada) e DOC_ID (identificação da imagem na referente ao AX, útil para garantir a integração de dados).

getImages

Nome: getImages

Inputs: Help_online_id, criterio.

Outputs: outputs(NOME, DESCRICAO, url), size, status.

Dependências: N/A.

Descrição: Serviço que dado um identificador de projecto e um critério devolve uma lista das imagens existentes na tabela IMAGENS_REFDATA cujo nome do ficheiro satisfaz o critério passado como input. Os campos de cada imagem que são devolvidos são o NOME_FICHEIRO (nome do file referente à imagem, por exemplo “teste.jpg”), a DESCRICAO (campo que permite aos utilizadores identificarem mais facilmente a imagem, sendo preenchido por um titulo ou resumo da imagem) e o url (localização da imagem no servidor, bastante útil para o módulo WYSIWYG).

getImagesForWysiwyg

Nome: getImagesForWysiwyg

Inputs: projecto.

Outputs: outputs(NOME, DESCRICAO, URL)[], status.

Dependências: N/A.

Descrição: Serviço que dado um identificador de projecto devolve uma lista das imagens existentes na tabela IMAGENS_REFDATA e correspondem ao projecto passado. Os campos de cada imagem que são devolvidos são o NOME_FICHEIRO (nome do file referente à imagem, por exemplo “teste.jpg”), a DESCRICAO (campo que permite aos utilizadores identificarem mais facilmente a imagem, sendo preenchido por um titulo ou

resumo da imagem) e o url (localização da imagem no servidor, bastante útil para o módulo WYSIWYG).

getLanguages

Nome: getLanguages

Inputs: projecto.

Outputs: locale(alpha2, label)[], status.

Dependências: N/A.

Descrição: Serviço que dado um identificador de um projecto devolve uma lista com as languages já activadas.

getLocaleDefault

Nome: getLocaleDefault

Inputs: N/A.

Outputs: propertyValue, status.

Dependências: N/A.

Descrição: Serviço que vai às extended settings do servidor IS e detecta a língua pré-definida.

getProjectName

Nome: getProjectName

Inputs: id.

Outputs: nome_projecto, status.

Dependências: N/A.

Descrição: Este serviço vai à tabela HELP_ONLINE, identifica a instância com o id transmitido como input e devolve o nome do projecto a que se refere determinado Help Online. Este serviço é útil na medida em que poderão ser criados diversos Help Online cada um associado a aplicações distintas, existindo desta forma tópicos, conteúdos e antecedências também distintas.

getProjects

Nome: getProjects

Inputs: N/A.

Outputs: Project(id, projecto, descricao)[], status.

Dependências: N/A.

Descrição: Serviço que devolve uma lista preenchida com todos os projectos existentes em base de dados na tabela HELP_ONLINE.

getTopicoAEditar

Nome: getTopicoAEditar

Inputs: projecto_id, topico_id, lingua.

Outputs: titulo_id, titulo, tipo, portlet, elemento, conteúdo, conteúdo_id, antecedente, antecedente_id, status.

Dependências: N/A.

Descrição: Serviço que devolve toda a informação indicada como output deste serviço relativamente a um determinado tópico de ajuda. Esta informação revela-se indispensável para preencher os dados de um tópico de ajuda quando este está a ser editado na interface de Administração.

getTopicoAjudaEditar

Nome: getTopicoAjudaEditar

Inputs: projecto_id, id_topico, lingua.

Outputs: dados_topico(id, titulo, tipo, portlet, elemento, antecedente, conteudo)[], status.

Dependências: N/A.

Descrição: Este serviço devolve o tópico de ajuda cujo projecto, a língua e o id correspondem ao inserido como input. Os campos de output deste serviço são preenchidos com dados recolhidos das tabelas TOPICOS_AJUDA, TOPICO_AJUDA_CONTEUDO e TOPICO_AJUDA_TITULO. Este serviço é executado no “initialize” da view EditarAjuda preenchendo todos os elementos do tópico de ajuda.

getTopicoName

Nome: getTopicoName

Inputs: id, lingua, PROJECTO.

Outputs: nome, status.

Dependências: N/A.

Descrição: Serviço que devolve o título de um determinado tópico de ajuda de um projecto. O comando em base de dados que devolve o resultado pretendido percorre três tabelas, HELP_ONLINE, TOPICOS_AJUDA e TOPICO_AJUDA_TITULO.

getTopics

Nome: getTopics

Inputs: id_projecto, língua.

Outputs: topicos_outputs(id, titulo, antecedente)[], status.

Dependências: N/A.

Descrição: Serviço que dado um identificador do projecto e uma determinada língua devolve uma lista preenchida com os tópicos do projecto informando para cada elemento o respectivo id, título e id do tópico antecedente. Percorre as tabelas HELP_ONLINE, TOPICOS_AJUDA e ANTECEDENCIAS para devolver o resultado solicitado. Este serviço adquire especial importância dado que é o responsável por alimentar as árvores de tópicos de ajuda existentes pelas diferentes views do projecto. Também é responsável por alimentar os arrays utilizados pelo método que constroem o breadcrumbs (sequência de tópicos) chamado pela pesquisa por tópico da view FrontOffice.

getURL

Nome: getURL

Inputs: nome, projecto.

Outputs: url, status.

Dependências: N/A.

Descrição: Serviço que devolve o url de uma imagem existente na tabela IMAGENS_REFDATA cujo nome e o projecto correspondem aos passados como input do serviço. Útil para pré-visualizar as imagens de um projecto em novas janelas despoletadas pelas views BackOffice e EditarAjuda.

insertBDHistory

Nome: insertBDHistory

Inputs: BDHelpOnline (Documento canónico cujo conteúdo representa fielmente todos os dados existentes em base de dados na altura em que o serviço é corrido).

Outputs: status.

Dependências: historyHandler (trigger que despoleta acção através do broker)

Descrição: Serviço que recebe como input um documento canónico que representa fielmente o estado actual da base de dados completa do Help Online e replica-a para o schema help_online_history. Esta acção fica a cargo do broker evitando desta forma que o utilizador tenha que aguardar o término da replicação da base de dados sempre que este serviço seja despoletado. O insertBDHistory tem uma acção preventiva uma vez que sempre que existem alterações no schema principal do Help Online é guardada uma cópia integral no schema de histórico.

insertImage

Nome: insertImage

Inputs: nome, descricao, url, help_id, docId, fileBytes

Outputs: status.

Dependências: N/A.

Descrição: Serviço despoletado quando é carregada uma nova imagem para o servidor onde se insere o Help Online. Sempre que é criada uma nova imagem por “File System” no servidor é necessário armazenar a sua localização na base de dados. Essa referência à imagem é criada na tabela IMAGENS_REFDATA. De forma a garantir a integração dos dados é possível passar uma cópia dos bytes da imagem para o servidor AX que por sua vez devolve um docId armazenado igualmente na IMAGENS_REFDATA.

pesquisaPalavraChave

Nome: pesquisaPalavraChave

Inputs: criterio, projecto_id, lingua.

Outputs: topicos(id, titulo, antecedente)[], status, size.

Dependências: N/A.

Descrição: Serviço que devolve uma lista de tópicos de um determinado projecto e língua composta apenas por elementos cujo título vai de encontro ao critério passado como input neste serviço. Este serviço adquire importância quando o utilizador efectua uma pesquisa por tópico na view de FrontOffice sendo-lhe transmitido um ou mais breadcrumbs (sequência de tópicos até ser atingindo o tópico que satisfaz o critério). O serviço getTopicos a par do pesquisaPalavraChave permite satisfazer os métodos responsáveis por criar os arrays que o breadcrumbs utiliza.

removeImage

Nome: removeImage

Inputs: GUID, docId, projecto.

Outputs: status.

Dependências: N/A.

Descrição: Serviço que dado o GUID de determinada imagem constante na base de dados, nomeadamente na tabela IMAGENS_REFDATA, executa as acções de delete correspondentes. Caso se justifique é efectuada uma tentativa de eliminação da imagem do servidor AX, responsável por garantir uma integração de dados. Este serviço é despoletado quando em BackOffice é pedido para ser removida uma imagem, tendo esta que ser eliminada não só por “File System” na pasta do servidor mas também a sua referência na base de dados e no AX.

removeLanguage

Nome: removeLanguage

Inputs: loc, projecto.

Outputs: status.

Dependências: N/A.

Descrição: Serviço despoletado na view BackOffice após um técnico indicar a remoção de uma língua activa. Este serviço recebe o identificador do projecto sobre o qual vai efectuar a acção e a língua a remover. Ao ser despoletado remove da base de dados todos os valores referentes a essa língua. Efectua assim uma acção de delete nas tabelas

TOPICOS_AJUDA, TOPICO_AJUDA_CONTEUDO, TOPICO_AJUDA_TITULO e ANTECEDENCIAS.

removerProjecto

Nome: removerProjecto

Inputs: id_projecto.

Outputs: status.

Dependências: N/A.

Descrição: Serviço despoletado na view de Administração. Este serviço recebe como input o identificador do projecto e remove todos os tópicos de ajuda, conteúdos, títulos e imagens associadas. São efectuadas acções de delete em todas as tabelas do schema Help Online de forma a que todos os dados inerentes a um determinado projecto sejam totalmente eliminados.

removeTopico

Nome: removeTopico

Inputs: id_topico, language, id_projecto.

Outputs: status.

Dependências: N/A.

Descrição: Serviço despoletado na view de Administração que dado os identificadores de um tópico de ajuda remove-o da base de dados. Além da acção de delete na tabela TOPICOS_AJUDA são igualmente eliminadas informações da tabela ANTECEDENCIAS e TOPICO_AJUDA_TITULO. No caso do conteúdo do tópico de ajuda que se pretende eliminar não estar associado a mais nenhum tópico, então neste caso é efectuada uma acção de delete na tabela TOPICO_AJUDA_CONTEUDO. Para os casos particulares em que o tópico que se pretende eliminar possui ramos (ou seja ele é antecedente de outros tópicos) então a tabela de ANTECEDENCIAS sofre update passando os tópicos imediatamente procedentes ao eliminado a ter como antecedente o mesmo do tópico eliminado.

setEstado

Nome: setEstado

Inputs: id, loc, estado, projecto.

Outputs: status.

Dependências: N/A.

Descrição: Serviço despoletado pela view de EditarAjuda e que efectua uma actualização no estado de um determinado conteúdo do tópico. Na prática este serviço permite transmitir para a base de dados as alterações entre activo e desactivo. Se o tópico é transmitido como activo então é efectuado um update na tabela TOPICO_AJUDA_CONTEUDO alterando-se o valor disabled para “0”, caso o tópico esteja activo este update é efectuado com o valor “1”. Esta alteração tem consequências imediatas para os utilizadores do portal dado que estes receberão informações distintas consoante o estado do tópico.

updateTopico

Nome: updateTopico

Inputs: id_topico, id_projecto, tipo_u, portlet_u, elemento_u, criar_novo_conteudo, id_novo_conteudo, desc_novo_conteudo, titulo_u, id_novo_antecedente, updated_by.

Outputs: status.

Dependências: N/A.

Descrição: Serviço despoletado pela view de Administração permitindo efectuar alterações à estrutura dos tópicos de ajuda de um determinado projecto. Através deste serviço é possível modelar todo um projecto, desde os títulos dos tópicos até à árvore de aplicação, títulos, conteúdos e toda a restante informação inerente a um determinado tópico. Os factos menos intuitivos deste serviço prendem-se com os problemas levantados por conteúdos associados, antecedências e línguas. Se um tópico for actualizado sendo indicado que este passará a ter um novo conteúdo então caso o antigo conteúdo associado deixe de ter tópicos associados é automaticamente eliminado. Este procedimento evita a existência de uma gestão complexa de conteúdos. Relativamente às antecedências, este serviço comporta-se da mesma forma que o removeTopics, ou seja, nos casos particulares em que o tópico que se pretende alterar a antecedência possui ramos (situações em que ele próprio é antecedente de outros tópicos) então os tópicos imediatamente procedentes ao alterado passam a ter como antecedente o antigo antecedente do tópico

alterado. Por fim se qualquer elemento estrutural do tópico de ajuda é alterado essas alterações repercutem-se ao nível de todas as línguas activas do projecto, por exemplo, se alterar alguma posição de um elemento da árvore de tópicos essa alteração é efectuada nas árvores das restantes línguas.

updateTopicoAjuda

Nome: updateTopicoAjuda

Inputs: id_projecto, id, língua, titulo, conteúdo, updated_by.

Outputs: status.

Dependências: N/A.

Descrição: Serviço despoletado pela view EditarAjuda quando o técnico efectua alterações a um determinado tópico previamente seleccionado. As alterações têm efeito directo sobre duas tabelas na base de dados: TOPICO_AJUDA_CONTEUDO e TOPICO_AJUDA_TITULO, possibilitando alterações ao nível do título, estado e conteúdo do tópico.