

Integração do iPortalDoc com Sistemas ERP

Rui Miguel da Silva Carvalho

Dissertação realizada no âmbito do
Mestrado Integrado em Engenharia Electrotécnica e de Computadores
sob orientação de
Professor Doutor José António Faria

(O Presidente do Júri, Professor Doutor José Machado da Silva)

Faculdade de Engenharia da Faculdade do Porto
Departamento de Engenharia Electrotécnica e de Computadores
Rua Roberto Frias, s/n, 4200-465 Porto, Portugal

Março de 2008

Resumo

O trabalho que se apresenta incidiu sobre o estudo e desenvolvimento de módulos de integração do sistema de gestão documental e *workflow* iPortalDoc com sistemas de gestão empresarial. Visou desenvolver uma ferramenta de configuração gráfica de apoio à criação de modelos de documentos funcionando de forma integrada com os vários sistemas.

Na primeira parte do documento apresenta-se um estudo sobre sistemas de gestão documental, sistemas de gestão de *workflow* e sistemas de gestão empresarial que permitem fazer um enquadramento do tema. Seguidamente foram apresentadas as principais tecnologias utilizadas no desenvolvimento deste trabalho, nomeadamente *Extensible Markup Language*, *Extensible Style Language* e *frameworks JavaScript*.

Após a apresentação dos sistemas e das tecnologias, apresentam-se o sistema de gestão documental e *workflow* iPortalDoc e o sistema operativo de suporte, IPBrick.

Na terceira parte apresentam-se as funcionalidades e a arquitectura da ferramenta de configuração gráfica de apoio à criação de modelos de documentos desenvolvida. Estes documentos são geridos pelo sistema de gestão documental e o ciclo de vida do mesmo é controlado pelo sistema de gestão de *workflow*.

Palavras-chave: modelos de documentos, relatórios, sistemas de gestão documental, sistemas de gestão de *workflow* e sistemas de gestão empresarial.

Abstract

This document presents the results of a project consisting on the study and development of a set of modules for the integration of the iPortalDoc document and workflow management system with enterprise management systems.

The project was focused on the development of a graphical configuration tool to support the creation of document models that should work together with the many information systems.

The first part of the document presents a state of the art survey on the information systems fields relevant for the project, namely on document management system, workflow management system, enterprise management systems, and on the main technologies employed during the project, namely Extensible Markup Language, Extensible Style Language, frameworks JavaScript will be introduced.

After this survey on systems and technologies, the iPortalDoc document management system and workflow, and the IPBrick operating system will be discussed.

The third part of the document discusses the functionalities, the architecture and the design of the document models configuration tools that were developed. These documents are managed by the documentation management system, and the life cycle is controlled by the workflow management system.

Key-words: document models, reports, document management system, workflow management system and enterprise management systems.

Agradecimentos

Começaria por agradecer à minha família em especial aos meus pais, irmã e meus avós pela forma insuperável como me acompanharam e ajudaram, e sem os quais este trabalho não teria sido possível.

À minha namorada pela compreensão, carinho, paciência e incondicional apoio que sempre demonstrou.

Um agradecimento especial ao meu orientador Prof. Doutor José António Faria pela sua disponibilidade e acompanhamento.

A todos os colaboradores da empresa iPortalMais, pela sua disponibilidade e apoio para a realização deste projecto.

Aos meus amigos que de uma forma directa ou indirecta me apoiaram na realização deste projecto.

A todos muito obrigado.

Para ser grande, sê inteiro: nada
Teu exagera ou exclui.
Sê todo em cada coisa. Põe quanto és
No mínimo que fazes.
Assim em cada lago a lua toda
Brilha, porque alta vive.

Fernando Pessoa

Conteúdo

1	Introdução	1
1.1	Objectivos	3
1.2	Estrutura	3
2	Arquitectura Geral e Principais Tecnologias	5
2.1	Gestão Documental	7
2.1.1	História	7
2.1.2	Sistemas de Gestão Documental	9
2.1.3	Classificação da Informação	10
2.1.4	Conceitos Aplicados na Gestão Documental	11
2.1.5	Principais Funcionalidades dos SGD	12
2.1.6	Benefícios dos SGD	14
2.2	<i>Workflow</i>	14
2.2.1	História	15
2.2.2	Sistema <i>Workflow</i>	15
2.2.3	Sistema de Gestão de <i>Workflow</i>	18
2.2.4	Consórcio de Gestão de <i>Workflow</i>	19
2.2.5	Características dos <i>Workflows</i>	21

2.2.6	Benefícios dos <i>Workflows</i>	22
2.2.7	Categorias de <i>Workflows</i>	22
2.2.8	Requisitos Típicos numa Aplicação <i>Workflow</i>	23
2.3	Sistema de Gestão Empresarial - ERP	24
2.3.1	Potencialidades dos Sistemas ERP	25
2.3.2	Implementação de um Sistema ERP	26
2.3.3	Pontos Críticos do ERP	27
2.4	Tecnologias	30
2.4.1	XML	30
2.4.2	XSL	35
2.4.3	<i>Frameworks JavaScript</i>	47
2.5	<i>Reporting Tools</i>	59
2.5.1	Comprar ou Desenvolver	59
2.5.2	Funcionalidades	60
2.5.3	Constituintes do Relatório	62
2.5.4	Ferramentas	62
3	Plataformas Tecnológicas	65
3.1	iPortlaDoc	65
3.1.1	Arquitectura	66
3.1.2	Segurança	66
3.1.3	Permissões	68
3.1.4	Modelos de Documentos	70
3.1.5	Tipos de Documentos	72
3.1.6	Introdução de Documentos	73

3.2	IPBrick	79
3.2.1	IPBrick.I	80
3.2.2	IPBrick.C	81
3.2.3	IPBrick.GT	83
3.2.4	IPBrick.KAV	84
3.3	Gestix Net	86
4	Apresentação	87
4.1	Especificações	87
4.1.1	Elementos	87
4.1.2	Propriedades dos Elementos	92
4.1.3	Configurações	95
4.1.4	Relatórios	100
4.2	<i>Design</i> da Ferramenta	102
4.2.1	Ficheiros Gerados	102
4.2.2	Criação de Documentos	105
4.2.3	Informação XML	105
4.2.4	Armazenamento da Informação	109
4.2.5	Modelo de Dados	110
4.2.6	Modelo de Camadas	111
4.2.7	Assunções	113
4.3	Implementação	114
4.3.1	Menu Campos	115
4.3.2	Menu Tabelas	116
4.3.3	Menu Pop-list	117

4.3.4	Menu Outros	119
4.3.5	Menu Template	120
4.3.6	Propriedades dos Elementos	121
4.3.7	Utilização dos Modelos	123
5	Conclusão	127
5.1	Perspectivas de Desenvolvimento	129

Lista de Figuras

2.1	Arquitectura geral	7
2.2	Classificação da informação	10
2.3	Exemplo de um <i>workflow</i> relativo a despesas	17
2.4	Conceitos associados aos WfMC [21]	19
2.5	Modelo de referência do WfMC [21]	20
2.6	Estrutura de um ERP [14]	25
2.7	Estrutura em árvore dum documento XML	31
2.8	Estrutura de um documento XML	32
2.9	<i>Document Type Declaration</i>	33
2.10	Troca de dados entre sistemas heterogéneos [4]	35
2.11	Exemplo de um XSLT	38
2.12	Transformação de um documento XML através de XSLT para HTML	39
2.13	Nós de um documento XML	39
2.14	Transformação de documentos XML [6]	42
2.15	Estrutura do XSL-FO	43
2.16	Sub-divisões da página	44
2.17	Exemplo do código XSL-FO	45
2.18	Transformação de um documento XML para PDF	46

2.19	Desenvolvimento total	47
2.20	Desenvolvimento utilizando bibliotecas	47
2.21	Desenvolvimento utilizando <i>frameworks</i>	47
2.22	Exemplo <i>Color Picket</i>	57
2.23	Exemplo <i>Date Picket</i>	57
2.24	Exemplo <i>Tabbed Pane</i>	58
3.1	Arquitetura do iPortalDoc	67
3.2	Hierarquia dos documentos	68
3.3	Perfis	70
3.4	Permissões nas directorias	71
3.5	Introdução de modelos de documentos	71
3.6	Tipos de documento	73
3.7	Exemplo de um <i>workflow</i> no iPortalDoc	75
3.8	Introdução de um documento	77
3.9	IPBrick.I inserida na LAN de uma empresa [22]	80
3.10	IPBrick.I na LAN de uma empresa com AD [22]	80
3.11	IPBrick.C numa DMZ com sistema de <i>firewall</i> [22]	82
3.12	IPBrick.C numa DMZ protegida por <i>firewall</i> [22]	82
3.13	IPBrick.KAV - <i>Appliance</i> de Segurança [22]	86
4.1	Elemento Editor de Texto	88
4.2	Restrições ao nível do posicionamento dos elementos	94
4.3	Posicionamento relativo dos elementos	95
4.4	Margens da página	96
4.5	Margens da página com cabeçalho e/ou rodapé	97

4.6	Cabeçalho e rodapé	98
4.7	Cabeçalho e rodapé com elementos	98
4.8	Geração de relatórios	102
4.9	Criação de documentos PDF	106
4.10	Criação dos ficheiros XML Formulário	107
4.11	Ficheiro XML Documento	108
4.12	Armazéns da aplicação	111
4.13	Modelo de dados	112
4.14	Arquitectura três camadas	113
4.15	Áreas de navegação	115
4.16	Elementos do menu Campos	116
4.17	Elementos do menu Tabelas	117
4.18	Elementos do menu Pop-list	117
4.19	Pop-list definida	118
4.20	Pop-list dinâmica	118
4.21	Elemento pop-list dinâmica - Formulário de pesquisa	118
4.22	Menu formato da página	119
4.23	Menu margens da página	119
4.24	Menu cabeçalho / rodapé da página	119
4.25	Configuração da base dados externa	120
4.26	Menu Template	121
4.27	Opções dos elementos	122
4.28	Menu Posição / Dimensão	122
4.29	Menu Alinhamento	122

4.30	Menu Fonte	123
4.31	Modelo de documento	124
4.32	Formulário de documento	124
4.33	Lista de documentos	125

Lista de Tabelas

2.1	Biblioteca vs <i>Framework</i>	48
2.2	<i>Frameworks</i>	50
2.3	Efeitos/Animações das <i>frameworks</i>	56
3.1	Campos de classificação entidades	78
3.2	Serviços da IPBrick.I	81
3.3	Serviços da IPBrick.C	83
3.4	Características da IPBrick.GT	84
3.5	Características da IPBrick.KAV	85
4.1	Campos pré-inseridos - Classificação	90
4.2	Propriedades dos Elementos	93

Lista de Acrónimos

SGDW: Sistema de Gestão Documental e *Workflow*

SGD: Sistema de Gestão Documental

SGE: Sistema de Gestão Empresarial

SGW: Sistema de Gestão de *Workflow*

ERP: *Enterprise Resource Planning*

PDF: *Portable Document Format*

XML: *eXtensible Markup Language*

SGML: *Standard Generalized Markup Language*

XSL: *eXtensible Style Language*

WfMS: *Workflow Management System*

WfMC: *Workflow Management Coalition*

WAPI: *Workflow API and Interchange Formats*

HTML: *HyperText Markup Language*

DTD: *Document Type Declaration*

XSD: *Schema Definition*

XSLT: *eXtensible Style Language Transformations*

XPath: *eXtensible Markup Language Path Language*

XSL-FO: *eXtensible Style Language - Formatting Objects*

DOM: *Document Object Model*

AJAX: *Asynchronous JavaScript and XML*

OLAP: *Online Analytical Processing*

OLTP: *On Line Transaction Processing*

Capítulo 1

Introdução

O presente trabalho foi desenvolvido na empresa iPortalMais - Soluções de Engenharia para Internet e Redes, Lda. Este enquadra-se num protocolo estabelecido entre a empresa e a Faculdade de Engenharia da Universidade do Porto, e teve início no mês de Setembro de 2007 com duração de um semestre.

Este trabalho compreende o desenvolvimento de um projecto que consiste na análise e desenvolvimento de módulos de integração do sistema de gestão documental e *workflow* iPortalDoc com sistemas de gestão empresarial - ERP. Pretende-se que seja desenvolvida uma ferramenta de fácil utilização para que esta possibilite o apoio à formatação de matrizes de documentos, ou seja, criação de formulários para posterior geração de documentos PDF. A ferramenta irá permitir a obtenção de relatórios com indicadores de gestão referentes aos processos geridos pelo SGDW e informação de gestão armazenada em ERP's com base na informação da sua base de dados. Os relatórios produzidos são posteriormente armazenados no sistema de gestão documental e associados ao respectivo *workflow*.

A interligação e integração das soluções de sistemas de gestão documentais com outras áreas aplicacionais é um passo importante no desenvolvimento das empresas. Garantir o fluxo de informação entre as diversas aplicações sem a necessidade de utilizar documentos físicos e introduzir a informação manualmente nas várias aplicações torna-

se uma solução necessária na vida actual das empresas. A redução do tempo associado ao tratamento da informação e a consequente libertação dos colaboradores para efectuar outras actividades, garantem o melhor desempenho das empresas [25].

O expediente administrativo de uma empresa processa-se normalmente através da entrada e saída de grandes volumes de documentação baseada em papel, utilizando diversos canais e formatos. A gestão e o encaminhamento de todo este volume de informação, muitas vezes crítica para o negócio, gera grande complexidade pela quantidade de cópias que origina e cria circuitos morosos de distribuição interna.

O iPortalDoc é um serviço de gestão documental e *workflow* para empresas e instituições que manifestem a necessidade de ter a gestão documental disponível na rede de comunicações e que seja de fácil utilização. O iPortalDoc permite, através da *web*, o acesso aos documentos e ficheiros sediados na empresa possibilitando a sua utilização e manutenção em qualquer parte do mundo através de uma simples ligação à Internet. Actualmente, as empresas estão imersas num mundo de ficheiros e documentos de origens e destinos variados, exigindo assim soluções rápidas para encontrar, tratar e arquivar os seus documentos. Desta forma, o sistema de gestão documental é um poderoso auxiliar na gestão corrente de qualquer empresa, sendo actualmente considerada uma ferramenta fundamental, independentemente da sua dimensão ou área de actividade [24].

A gestão documental, quer seja electrónica quer seja em arquivo de papel, está presente em todas as empresas. Desta forma, a implementação da gestão documental é tão indispensável como a implementação de um sistema ERP.

Com o aumento da complexidade e volume de negócio, algumas das respostas sobre o momento actual da empresa são até impossíveis. É então que surge a necessidade de informatizar a organização, e a implementação de um ERP permite isso mesmo: conhecer melhor a empresa e disponibilizar a informação certa no momento certo. Um sistema ERP é uma ferramenta de trabalho integrada, e, assim, as empresas deixam de sentir a necessidade de trabalhar com várias ferramentas e interfaces. Isto porque, com um sistema integrado, as empresas podem finalmente realizar o cruzamento de dados de que necessitam e que realmente lhes acrescenta valor, ganhando desta forma poder de reacção

às mudanças que actualmente todas as empresas enfrentam no seu mercado.

1.1 Objectivos

O principal objectivo deste trabalho é desenvolver uma ferramenta capaz de produzir modelos de documentos configurados pelo utilizador, sendo que estes modelos apresentam duas perspectivas de funcionamento.

A ferramenta permite obter relatórios com indicadores de gestão referentes aos processos geridos nos sistemas ERP através do conteúdo da base de dados e nesta perspectiva apresenta-se como uma alternativa às ferramentas de geração de relatórios (*reporting tools*).

Numa segunda perspectiva a ferramenta permite a criação de simples documentos com informação fornecida pelo utilizador. Desta forma, o utilizador preenche formulário e, com a informação do mesmo é criado o documento.

A ferramenta funciona de forma integrada com o iPortalDoc, apresenta-se como uma funcionalidade incorporada neste. Sempre que sejam utilizados os modelos de documentos produzidos por esta funcionalidade, os documentos são introduzidos no sistema de gestão documental e associados ao respectivo *workflow*, iPortalDoc.

1.2 Estrutura

O presente relatório está organizado em cinco capítulos.

Neste primeiro capítulo é apresentada uma introdução e enquadramento do trabalho bem como são apresentados os principais objectivos do mesmo.

No capítulo 2 é apresentada a arquitectura geral de funcionamento e as principais tecnologias presentes no desenvolvimento e funcionamento da ferramenta.

No capítulo 3 são apresentadas as plataformas tecnológicas de suporte ao funcionamento da ferramenta desenvolvida.

No capítulo 4 é apresentado a ferramenta desenvolvida, apresentando as principais especificações e a sua implementação.

Para finalizar, no capítulo 5, é apresentada uma síntese com as principais conclusões e perspectivas futuras de desenvolvimento.

Capítulo 2

Arquitectura Geral e Principais Tecnologias

Sempre que as necessidades da organização assim o justifiquem, deverá ser possível criar modelos de documentos que poderão ser muito úteis, por exemplo para actas de reunião, facturas, faxes, relatórios entre outros. A ferramenta desenvolvida relaciona-se, sobretudo, com o aspecto gráfico do documento, desta forma o SGDW iPortalDoc deverá disponibilizar uma ferramenta de configuração gráfica aos seus utilizadores.

A ferramenta de apoio à criação de modelos de documentos tem duas perspectivas de utilização. Permite processar relatórios estatísticos através da base de dados dos sistemas ERP permitindo ultrapassar algumas limitações deste tipo de sistemas. Os sistemas ERP apresentam uma solução genérica, sendo implementados de forma diferente em cada organização, e, desta forma, os relatórios podem não estar adequados a todas as necessidades da organização.

Numa segunda perspectiva, a ferramenta permite ao utilizador responder a um formulário, e com a informação introduzida é gerado um documento PDF, sendo armazenado e associado ao respectivo *workflow* no SGDW. Assim, a ferramenta pode ser utilizada para produzir os modelos de faxes, correspondência ou até ordens de compra, entre outras.

Para a utilização da ferramenta pode identificar-se duas fases distintas. A primei-

ra fase consiste no desenvolvimento dos modelos de documentos e a segunda permite a utilização destes modelos de forma integrada com o restante SGDW iPortalDoc.

Após o desenvolvimento destes modelos são gerados um conjunto de ficheiros que permitem fazer a integração entre os vários sistemas a interligar. Na figura 2.1 está representada a arquitectura geral deste processo.

Quando é introduzido ou editado um documento é apresentado um formulário ao utilizador. O formulário pode reunir informação de duas fontes de dados, podem ser apresentados dados com origem na base de dados do ERP e pode ainda ser introduzida informação pelo utilizador.

Quando a informação estiver completamente reunida no formulário será armazenada no formato XML. A informação poderá ser também armazenada na base de dados, e quando esta situação se verifica é possível ao utilizador configurar novos modelos de documentos utilizando a informação armazenada.

O processo de criação do documento PDF é efectuada utilizando o ficheiro XML anteriormente armazenado. O ficheiro XML é submetido a uma transformação utilizando um estilo XSL e assim é gerado o documento PDF.

O documento PDF é armazenado no SGD efectuando a sua gestão e por fim é associado ao respectivo *workflow*, permitindo a gestão do fluxo do respectivo documento.

Nas próximas secções apresentam-se as tecnologias presentes no desenvolvimento deste projecto. Efectua-se uma revisão conceptual sobre sistemas de gestão documental, seguido pelos sistemas de gestão de *workflow* e sistemas de gestão empresarial. Após a apresentação destes sistemas apresenta-se as tecnologias XML e XSL presentes no processo de gestão e apresentação da informação.

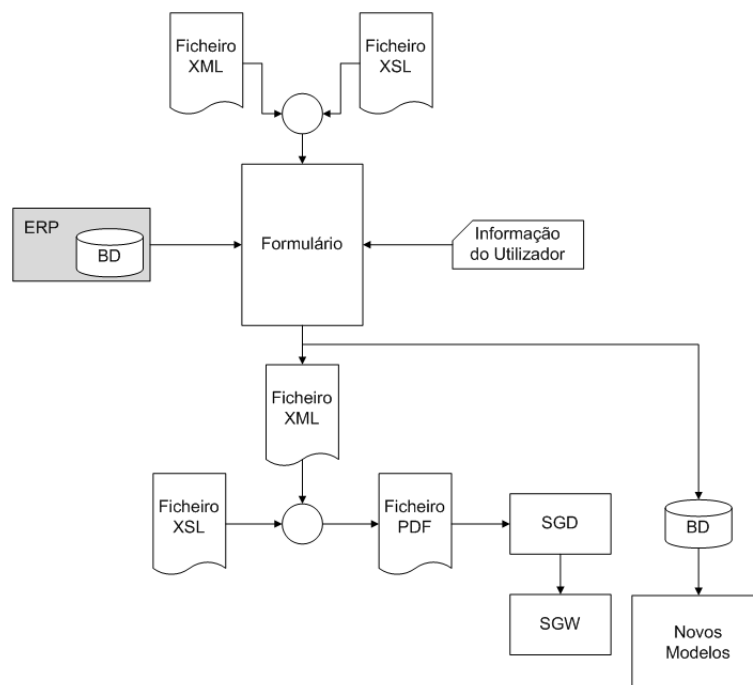


Figura 2.1: Arquitectura geral

2.1 Gestão Documental

Esta secção tem como objectivo apresentar alguns conceitos relativos à gestão documental, sendo estes aspectos organizados em sub-secções. Assim, a sub-secção 2.1.1 apresenta uma introdução à história da gestão documental; a sub-secção 2.1.2 faz uma abordagem aos sistemas de gestão documental; a sub-secção 2.1.3 apresenta as finalidades e o método de classificar a informação; a sub-secção 2.1.4 apresenta um conjunto de conceitos aplicados à gestão documental; a sub-secção 2.1.5 apresenta um conjunto de funcionalidades dos sistemas de gestão documental e a sub-secção 2.1.6 lista os principais benefícios destes sistemas.

2.1.1 História

A informação tem adquirido ao longo do tempo uma grande importância principalmente após a segunda guerra mundial. A partir da segunda metade do século XIX, a

arquivologia desenvolveu-se como uma disciplina e o seu aspecto mais importante pode ser considerado a gestão de documentos. Os princípios de gestão resultaram, sobretudo, da necessidade de se racionalizar e modernizar as organizações.

Embora a sua concepção teórica e aplicabilidade se tenha desenvolvido após a segunda guerra mundial, a partir do E.U.A. e Canadá, a gestão de documentos teve origem no final do século XIX, em função dos problemas então detectados nas administrações públicas destes dois países, no que se referia ao uso e gestão da informação. Na primeira metade deste século, criaram-se comissões governamentais nos E.U.A. e no Canadá, com vista a encontrar soluções para a melhoria dos padrões de eficácia na utilização dos documentos por parte da administração pública. Importa referir que, durante este período, as instituições arquivísticas públicas caracterizavam-se pela sua função de apoio à pesquisa, comprometidos com a conservação e acesso aos documentos considerados de valor histórico. A esta concepção opunha-se a concepção de documentos administrativos, cujos problemas eram considerados da responsabilidade exclusiva dos órgãos da administração pública que os produziam e utilizavam.

Conforme mencionou Ricks [39], num trabalho apresentado no VIII Congresso Internacional de Arquivos, realizado em Washington, em 1976, a gestão de documentos criou uma maior consciência do significado dos documentos, independentemente do seu suporte, e da importância da necessidade de conservação. As instituições arquivísticas públicas, particularmente os arquivos nacionais dos E.U.A. e do Canadá, adquiriram uma nova configuração, assumindo também a função de órgão de apoio à administração pública, com a competência de orientar programas de gestão de documentos nos diversos organismos governamentais. Além disso, actualmente elas dispõem de considerável prestígio e de maiores orçamentos, uma vez que foi reconhecido que, como instituições, economizam mais dinheiro do que gastam, em resultado das suas actividades de gestão de documentos. Por exemplo, a rede de arquivos intermediários regionais norte-americanos permite aos cofres públicos uma poupança de cem milhões de dólares por ano [11].

2.1.2 Sistemas de Gestão Documental

A importância da informação para as organizações é universalmente aceite, constituindo, senão o mais importante, pelo menos um dos recursos mais importantes cuja gestão e aproveitamento estão directamente relacionados com o sucesso. A informação também é considerada e utilizada em muitas organizações como uma estrutura e um instrumento de gestão. Portanto, a gestão de uma organização requer uma percepção objectiva e precisa dos valores da informação e do sistema de informação.

Estudos efectuados apontam que, o volume de informação não estruturada produzido pelas organizações cresce de 65% a 200% por ano, dependendo do sector em causa. Estudos patrocinados por empresas da área de gestão documental e *content management* mostram que cada pessoa produz, em média, 800MB de dados a cada ano (valores relativos a 2002) [25].

A esta grande quantidade de informação soma-se, ainda, a incapacidade das empresas classificarem eficazmente e a dificuldade em perceberem se um determinado documento possui informação crítica ou se é apenas mais um ficheiro a sobrecarregar sistema. Também desconhecem se se trata ou não da última versão do ficheiro ou quem teve acesso ao documento. Resumindo: não se conhece o ciclo de vida da sua informação crítica, e isto tem custos e comporta riscos. É por isso que os sistemas de gestão documental fazem todo o sentido. São, aliás, um passo inevitável na modernização das empresas [25].

Identificar as vantagens competitivas de uma organização que gere os seus documentos de forma automatizada, comparativamente com outra que os gere de forma tradicional, torna-se cada vez mais evidente. Factores como a organização interna da empresa, os valores poupados em papel, e a rapidez na obtenção da informação, começam a diferenciar estes dois tipos de empresas. O segredo da percepção da mais valia que tem, quem opta por um sistema de gestão documental, está em entender como um destes sistemas resolve problemas, transformando risco em segurança, tempo perdido em produtividade, desorganização em organização, desmotivação em vontade e custos em proveitos [9].

A gestão documental é, actualmente, o meio mais eficaz para gerir documentos.

Entende-se por documentos todo o tipo de suporte utilizado para manter e divulgar informação. Estes (os documentos) podem ser do tipo tradicional, em papel, ou electrónicos contendo ambos na sua essência, o mesmo: informação. Face à especificidade de cada um dos tipos, devem ser tratados de forma própria, salvaguardando em ambos os casos o que é importante, a informação neles contida.

Através destas soluções, é possível consultar rapidamente qualquer informação de um documento, e colocá-la em contexto com o processo de negócio em que participa. As soluções de gestão documental são frequentemente procuradas para suportar o arquivo documental associado a processos onde o volume de documentos é significativo.

2.1.3 Classificação da Informação

A informação tem duas finalidades: para conhecimento dos ambientes interno e externo de uma organização e para actuação nestes ambientes [7]. A classificação deve ser realizada em função do papel que a informação pode desempenhar nas actividades de uma organização (informação crítica, mínima, potencial, sem interesse), conforme é apresentado na figura 2.2.

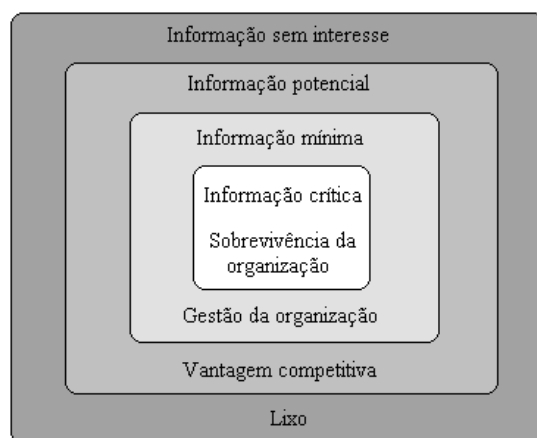


Figura 2.2: Classificação da informação

A importância da informação pode conduzir algumas organizações a obterem excesso de informação tornando-se mais difícil de gerir. As organizações devem concentra-

rem-se na manutenção da informação crítica, mínima e potencial. Contudo, esta operação é muito delicada, pois a classificação de uma dada informação, em particular, numa destas classes poderá ser um problema de difícil resolução.

Para minimizar a dificuldade na classificação, é necessário compreender um outro princípio, o do valor da informação. Em Saracevic [40] refere-se que a informação possui uma variedade de conotações em diferentes campos. Em alguns campos, incluindo a ciência da informação, a noção de informação está geralmente associada às mensagens. Nesse sentido, existe um elevado número de interpretações que são assumidas em diferentes abordagens teóricas e práticas para o tratamento da informação.

No contexto de uma organização, a informação deve atender às necessidades dos diversos níveis administrativos. Segundo Chiavenato [8], as organizações dividem-se em três níveis organizacionais, qualquer que seja o sector ou tamanho da organização:

- Operacional: relaciona-se com os problemas de desempenho e dirigido para as exigências impostas pela natureza da tarefa técnica;
- Intermediário: gerir particularmente as actividades do nível operacional, mediando as fronteiras ambientais e administrando as tarefas técnicas que devem ser desempenhadas, escala de operações, entre outras;
- Institucional: constitui-se na fonte do significado e da legitimação que possibilita a consecução dos objectivos organizacionais.

2.1.4 Conceitos Aplicados na Gestão Documental

A gestão documental electrónica é um processo abrangente que é originado aquando da recepção de um documento, e que implementa os seguintes conceitos:

- Digitalização: tem como objectivo digitalizar os documentos em papel;
- Armazenamento: catalogação e categorização dos documentos electrónicos. Esta fase é em todo equivalente ao processo de arquivo físico mas usufruindo dos benefícios das tecnologias de informação;

- *Workflow*: nesta etapa define-se os vários estados pelos quais um documento circula, incluindo publicação, aprovação, distribuição e reencaminhamento ou destruição (ver secção 2.2);
- Pesquisa: por fim, deverá ser implementado um motor de busca potente permitindo encontrar os documentos introduzidos.

2.1.5 Principais Funcionalidades dos SGD

Os SGD devem oferecer aos seus utilizadores um conjunto de funcionalidades a nível de utilização, encaminhamento, segurança, e administração. Em Pedro e Sezinando [41] apresentam-se estas funcionalidades.

Utilização e Encaminhamento

Mediante o uso de SGD, facilmente se efectua algumas das seguintes operações:

- Formatação de matrizes de documentos (modelos de documentos) sem exigência de instruções nem de programação;
- Criação de documentos com referência única e sua validação, nomeadamente tipo, designação, assunto, autor, classificação, numeração, versão, data de criação e revisão, encaminhamento, impressão e arquivo;
- Indexação das pastas e dos documentos por taxionomia hierárquica com o mínimo de três níveis, desenvolvida em função dos temas;
- Controlo de versões dos documentos com revisão dos seus atributos;
- Funcionalidades de trabalho colaborativo;
- Pesquisa e recuperação de informação por atributos ou por conteúdo, em todo o ciclo de vida dos documentos, garantindo o seu valor probatório;

- Encaminhamento e rastreabilidade de documentos criados ou importados, com inserção de comentários, pareceres e decisões, podendo as assinaturas manuscritas neles serem inseridas;
- Notificações de encaminhamentos com emissão de alertas sobre prazos limite;
- Registo, digitalização e arquivo de documentos recebidos e emitidos;
- Capacidade de integrar, importar e exportar conteúdos de diversos tipos, formatos, produtos e ambientes, nomeadamente texto, imagem, folhas de dados, gráficos, áudio, vídeo, CRM, ERP, correio electrónico, fax e documentos *web*;
- Impressão dos documentos em papel ou gravação de CD-ROM, DVD, ou outro suporte digital actual.

Segurança

Com um SGD, a confidencialidade da informação e controlo de acesso a dados e documentos está salvaguardada:

- Possibilidade de comunicação de dados encriptados e segurança através de assinaturas electrónicas e certificação cronológica;
- Segurança do sistema, confidencialidade da informação e controlo de acessos a dados e documentos, com definição de perfis de utilizadores.

Administração

Um SGD deverá garantir funções de administração, nomeadamente a alteração de matrizes, taxionomias e perfis de acesso, assim como métricas de informação de todos os documentos tratados, bem como tempos de tratamento e respectivo tratamento estatístico periódico. Deverá estar garantido também o uso de rotinas de auditoria, bem como interfaces parametrizáveis amigáveis para o utilizador.

2.1.6 Benefícios dos SGD

De acordo com Pedro e Sezinando [41], os principais benefícios de um SGD são os seguintes:

- Melhoria de acessos, da precisão e velocidade dos fluxos de informação;
- Melhoria da produtividade através da partilha de informação precisa entre utilizadores distintos;
- Menor gasto de tempo na procura de documentos críticos;
- Garantia de informação atempada sobre prazos a cumprir;
- Controlo de acessos a documentos críticos;
- Redução de custos e espaços de armazenamento;
- Restrição dos arquivos pessoais de cópias;
- Melhoria na tomada de decisão no tempo certo com os documentos necessários;
- Integração operacional de documentos de múltiplos formatos, nomeadamente texto, imagem, folhas de dados, gráficos, áudio, vídeo, correio electrónico e documentos *web*.

2.2 Workflow

Nesta secção apresenta-se um resumo de diversos conceitos relativos aos sistemas de gestão de *workflow*. Na sub-secção: 2.2.1 é apresentada a origem destes sistemas. Na sub-secção 2.2.2 efectua-se uma abordagem aos sistemas de *workflow*. A sub-secção 2.2.4, por sua vez, apresenta o modelo de referência proposto pelo WfMC. A sub-secção 2.2.5 apresenta as principais características/elementos dos *workflows*, a sub-secção 2.2.6 lista um conjunto de benefícios associados aos *workflows*, e a sub-secção 2.2.7 apresenta as categorias destes sistemas. Para finalizar, a sub-secção 2.2.8 apresenta os requisitos de um sistema *workflow*.

2.2.1 História

Os sistemas de *workflow* são bastante antigos, no entanto apenas nas últimas décadas começaram a ser populares, devido à necessidade das organizações automatizar os processos de negócio.

Historicamente, os sistemas de gestão de *workflow* têm a sua origem na automação do trabalho [31]. Eles eram, no entanto, monolíticos por natureza, de modo que as políticas de negócio e acesso às informações estavam rigidamente definidas nas aplicações. Além disso, estes sistemas quase sempre se encontravam isolados, sem ligação alguma com outros sistemas de *workflow* ou aplicações genéricas.

A primeira geração dos sistemas de *workflow* compreendeu, segundo descrito em Du e Elmagarmid [3], aplicações monolíticas de uma área de domínio em particular. A segunda geração dividiu os sistemas em componentes diversos, mas ainda os deixou fortemente dependentes das aplicações. A terceira geração apresentou máquinas de *workflow* genéricas que forneciam uma infra-estrutura robusta para *workflows* orientados à produção. Segundo os autores, uma quarta geração seria a actual, a de sistemas de *workflows* que oferecem uma gama diversa de serviços.

Com a evolução natural dos sistemas, tornou-se necessário separar processos de negócio e componentes de *workflow* das aplicações existentes, visando aumentar a flexibilidade e o poder de manutenção destes sistemas [3].

Actualmente os *workflows* têm tido um papel muito importante, conseguindo resolver os problemas dos processos e automatizar os mesmos a todos os níveis das organizações.

2.2.2 Sistema Workflow

A utilização da metodologia *workflow* na perspectiva da gestão de *workflows* visa automatizar a execução de processos de negócio, de forma parcial ou na sua totalidade. Neste âmbito, a execução dum processo de negócio pode envolver diversos sistemas li-

gados, bases de dados e aplicações, permitindo também a interacção com os utilizadores. Tradicionalmente, a implementação de um *workflow* é extremamente difícil de realizar devido à necessidade de integrar com diversas aplicações distintas e interacção com várias pessoas.

Nas empresas existem processos que são repetidos consecutivamente, e que possuem documentos associados a cada uma das suas fases. Um processo de *workflow* é um processo que pode ser pré-parametrizado, que tem responsáveis por cada um dos seus estados e no qual existem documentos que circulam entre os intervenientes. Um exemplo de um *workflow* são os processos de documentos que passam sempre pelos mesmos departamentos e de forma sequencial, como são os processos de aprovação de compras, onde um pedido passa por vários utilizadores (colaborador, responsável, sector financeiro, entre outros) [9].

Desta forma, deixam de existir documentos duplicados, os documentos deixam de se perder, e a empresa começa a ter percepção de quanto tempo cada documento está na posse de cada pessoa, e quanto tempo demora em média cada processo. Assim, a velocidade dos processos aumenta significativamente e o desempenho da empresa melhora consequentemente. A concepção, desenvolvimento, aprovação de factura e aprovação de despesas, são alguns dos exemplos de processos de *workflow*. Na figura 2.3 está representado um simples *workflow* relativo a despesas. No diagrama apresentado pode identificar-se três estados:

- Introdução da despesa;
- Aprovação da despesa. Neste estado ocorrerá uma de duas transições possíveis, sendo que se a despesa for aprovada o próximo estado será o pagamento da despesa. Caso contrário voltará ao estado inicial, introdução da despesa.
- Pagamento da despesa;

Workflow é definido como um conjunto de tarefas organizadas para realizar um processo, quase sempre de negócio. Essas tarefas podem ser executadas por um ou mais sistemas de computador, por um ou mais agentes humanos, ou então por uma

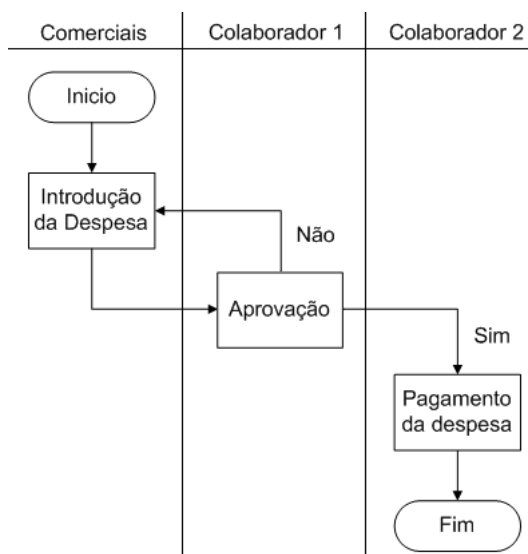


Figura 2.3: Exemplo de um *workflow* relativo a despesas

combinação destes. A ordem de execução e as condições pelas quais cada tarefa é iniciada também estão definidas no *workflow*, sendo que o mesmo é capaz ainda de representar a sincronização das tarefas e o fluxo de informações [32].

Um sistema de *workflow* pode ser interpretado como um sistema de informação responsável pela transição de documentos de um utilizador para outro de acordo um conjunto de regras predefinidas no negócio em causa [15] [28].

Actualmente, com a necessidade cada vez maior da informatização das organizações, a utilização de modelação de processos de negócio através de alguma ferramenta de *workflow* tornou-se uma prática bastante comum.

As organizações de vários sectores, lutam para desenvolver fluxos de trabalho automatizados que se adaptem prontamente às mudanças e aproveitem ao máximo todos os recursos disponíveis ou necessários.

Vários processos presentes no dia-a-dia das empresas podem ser automatizados, desde a solicitação e aprovação de férias, viagens e compra de produtos, até à definição das actividades que um colaborador deve desenvolver dentro de um determinado projecto.

A capacidade de um sistema de informação automatizar determinadas tarefas numa organização é crucial para a adaptação às necessidades de negócio. A metodologia *workflow* permite a combinação de regras associadas a processos de negócio, necessária à execução dos processos de negócio. Esta tecnologia deve possibilitar a modelação, gestão e monitorização dos fluxos de informação e trabalho associados aos processos de negócio.

A execução de processos de negócio de sistemas empresariais é importante e significativa na medida em que, a partir de um modelo de processos de negócio de uma empresa podem gerar automaticamente um conjunto de tarefas bem definidas. Utilizando uma ferramenta de *workflow* é possível automatizar algumas destas tarefas, permitindo também a monitorização do fluxo de informação ou do estado do processo de negócio.

2.2.3 Sistema de Gestão de *Workflow*

Um conceito importante é o de sistema de gestão de *workflow*, este proporciona a automação de um processo de negócio, gerindo a sequência de actividades de trabalho e seleccionando os recursos humanos e/ou electrónicos apropriados associados com os vários passos da actividade.

Por definição, pode dizer-se que o WfMS é um sistema que define, gere e executa completamente *workflows* através da execução de *software* cuja ordem de actividades é dirigida por uma representação da lógica do *workflow* no computador. As ferramentas de WfMS possibilitam um aviso prévio de acontecimentos/situações/actividades importantes para o utilizador. Uma vez que um processo é definido, um WfMS certifica-se de que as actividades ocorram numa sequência própria, e que os utilizadores sejam informados para que possam executar as suas tarefas. Por exemplo, em vez da informação de que parte de um trabalho foi concluída, para posteriormente se realizar a outra parte desenvolvida por outrem, passar oralmente entre pessoas, o WfMS faz uma notificação automática e imediatamente [32].

2.2.4 Consórcio de Gestão de *Workflow*

Em Agosto de 1993 foi criado o mais importante dos consórcios ligados aos estudos de sistemas de gestão de *workflow*: o WfMC (*Workflow Management Coalition*).

De acordo com o WfMC, *workflow* é a automação de um processo de negócio, na totalidade ou em parte. Conforme o WfMC, um sistema de gestão de *workflow* é um sistema que define, gere e executa *workflows*, cuja ordem de execução é dirigida por uma representação em computador da sua seriação lógica [21].

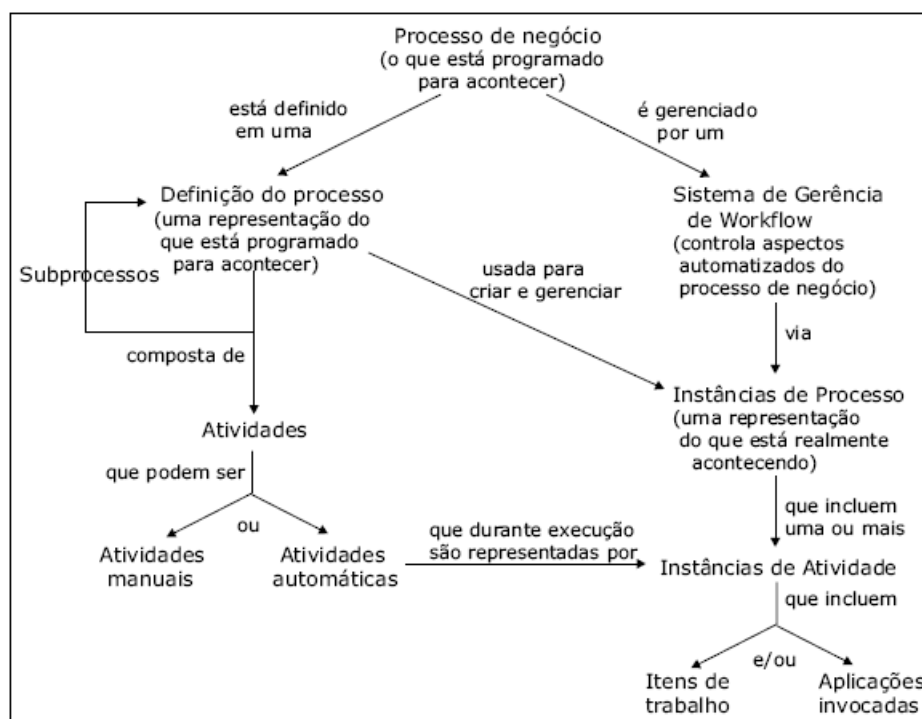


Figura 2.4: Conceitos associados aos WfMC [21]

De acordo com a figura 2.4, um processo de negócio é determinado por uma definição de processo e é composto por um conjunto de actividades, manuais ou automáticas, ou por outras definições de processos. Um processo é gerido pelo sistema de gestão de *workflows*. Assim como actividades automáticas e processos são representados em tempo de execução por instâncias. Instâncias de processo são compostas pelas instâncias das actividades que dele fazem parte. Estas actividades podem incluir itens de

trabalho ou aplicações invocadas.

Cada actividade num processo pode ser definida através de um modelo abstracto, que é basicamente uma máquina de estados cujo comportamento é dado pelas transições definidas entre eles. A estrutura de uma tarefa, no entanto, pode ser directamente afectada pelas características do utilizador que vai executá-la. O comportamento transaccional do utilizador, ou seja, o modo como ele executa as transacções que envolvem as tarefas do processo, determina o diagrama de estados da tarefa que nele executa. As tarefas podem estabelecer comunicação entre elas através das variáveis locais ao *workflow*. O fluxo de dados entre as sub-transacções (e portanto o seu relacionamento) é determinado através da associação de parâmetros de entrada e de saída às tarefas.

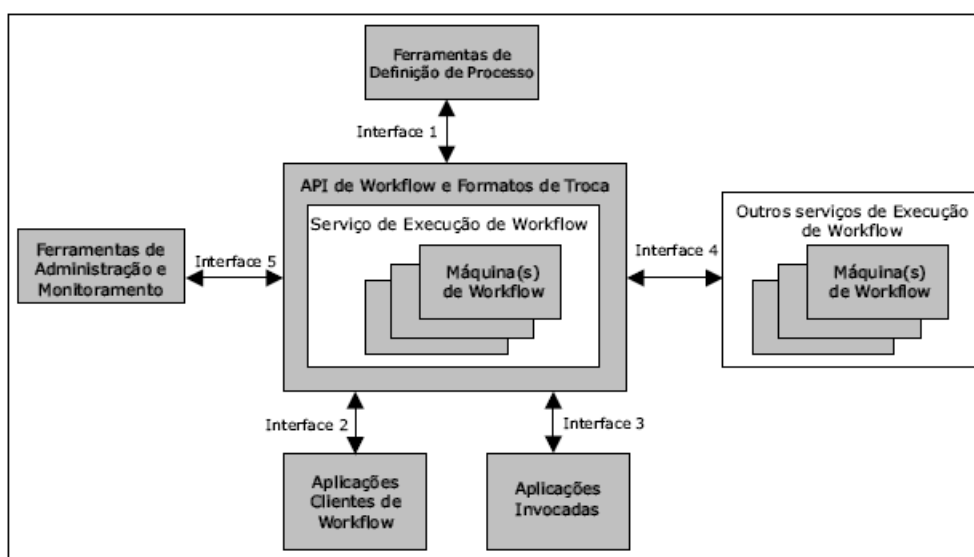


Figura 2.5: Modelo de referência do WfMC [21]

Os conceitos pertinentes aos sistemas de gestão de *workflow* foram utilizados posteriormente pelo WfMC para a criação do modelo de referência para sistemas de *workflow* [21]. Como está representado na figura 2.5, este modelo de referência identifica cinco componentes apresentados de seguida:

- Interface 1 - Ferramentas de Definição de Processos: especifica um padrão para a interface entre ferramentas de definição de processos e os serviços de execução de

workflow;

- Interface 2 - Aplicações Clientes de *Workflows*: define padrões para que o serviço de execução de *workflow* mantenha as tarefas que os aplicativos de *workflow* oferecem ao utilizador final. Para essa interface foram definidas operações dentro do escopo da WAPI (*Workflow API and Interchange Formats*) [45]. A WAPI pode ser considerada um conjunto de construtores através dos quais os serviços do sistema de *workflow* podem ser acedidos e os quais controlam as interacções entre a máquina de *workflow* e os outros componentes do sistema.
- Interface 3 - Aplicações Invocadas: define um padrão de interface para permitir que o serviço de execução de *workflow* utilize várias aplicações. Por exemplo, serviços de fax, correio electrónico, ou outras aplicações. Para dar suporte a esta interface, foram definidas algumas operações dentro da WAPI, tais como as operações de iniciar, suspender e abortar tarefas.
- Interface 4 - Outros Serviços de Execução de *Workflow*: define uma variedade de modelos que podem interagir com produtos de fabricantes diversos e os padrões para cada um;
- Interface 5 - Ferramentas de Administração e Monitorização: definem um padrão para funções de monitorização e controlo de processo. O WfMC apresenta os métodos de acesso da WAPI projectados para estas funções de gestão, tais como delegação e suspensão de privilégios a utilizadores e grupos, entre outras funções [45].

2.2.5 Características dos *Workflows*

Qualquer *workflow* é constituído por:

- Caminhos (*Routes*): caminhos entre conjuntos de objectos criando o *workflow*, podendo estes ser lineares, circulares ou paralelos;
- Regras (*Rules*): definem condições necessárias para permitir a transição de estado;

- Papéis (*Roles*): definem as funções das pessoas ou programas envolvidos no *workflow*;
- Tarefas (*Task*): acções que são conduzidas pelo *workflow*, e que podem ser efectuadas pelos utilizadores ou programas definidos no caminho.

2.2.6 Benefícios dos *Workflows*

Segundo Turner e Hall [18] os benefícios dos *workflows* são vários, entre eles destacam-se:

- Eliminação da documentação em papel;
- Simplificação dos formulários previstos;
- Acesso remoto;
- Arquivo e recuperação de informações simplificados;
- Habilidade de rapidamente trilhar as informações submetidas;
- Possibilidade de ter conhecimentos dos responsáveis de cada tarefa do processo;
- Aumento no tempo de linhas de informação.

2.2.7 Categorias de *Workflows*

Segundo Atlee [5], as tecnologias de *workflow* podem ser divididos em três categorias gerais:

- *Document routing*: estabelece o fluxo de informação e faz o encaminhamento dos mesmos;
- *Ad-hoc*: ferramentas de *groupware*. Nesta categoria, não existe uma estrutura pré-definida para o processo, ou esta estrutura pode ser modificada em tempo de

execução. Fornece a gestão de *workflow* através de templates ou formulários baseados em mensagens. O fluxo é gerido pelo servidor de encaminhamento de mensagens. Por exemplo: criação de documentos, desenvolvimento de *software*, requisições de viagem, campanha de *marketing* para lançamento de produto, entre outros;

- Automação de processo de negócios: sistema para definir processos de negócios e implementação dos mesmos em *software*.

2.2.8 Requisitos Típicos numa Aplicação *Workflow*

De acordo com Hatoun [33], os requisitos típicos numa aplicação baseada em *workflow* normalmente incluem:

- Aptidão para tomar decisões consoante as regras do negócio. Isto abrange regras simples, tais como decisões tipo sim ou não, baseadas numa caixa de validação, e regras mais complexas, tais como análises de conjuntos de dados para obter uma decisão não definitiva mas perto da decisão final;
- Formas de comunicar com outras aplicações e outros sistemas externos ao *workflow*. Por exemplo, um pedido é recebido por parte de outra aplicação, requerendo comunicações utilizando *web services* ou outras tecnologias;
- Métodos de interacção com pessoas. Um utilizador deve ter a possibilidade de aprovar ou validar algumas decisões. Deve, assim, existir uma interface que permita a interacção de um utilizador com a aplicação;
- Possibilidade de manter um estado durante a existência do *workflow*, especialmente quando pessoas estão envolvidas. Isto porque o desenvolvimento do *workflow* até este estar finalizado pode ser um processo bastante moroso, e um utilizador pode ir de férias ou de fim-de-semana interrompendo este mesmo desenvolvimento. Assim, O sistema deve de ser escalável de modo a ter a possibilidade de desactivar um *workflow*, mantendo o seu estado, e depois ter a capacidade de reactivar o *workflow*, retomando o seu estado e fluir para o passo seguinte.

2.3 Sistema de Gestão Empresarial - ERP

Actualmente, as empresas não podem ficar em atrasado relativamente aos concorrentes directos. Estes últimos anos têm impulsionado todo o mercado de negócios de uma forma dinâmica e competitiva, e todas as organizações enfrentam hoje novos mercados, novos concorrentes e consumidores, cada vez mais informados e exigentes.

As organizações estão em constante reorganização, com muitas integrações e actualizando as suas políticas e processos de produção, de forma a dar resposta aos concorrentes e às novas leis do mercado. As tecnologias de informação e processos de re-engenharia empresarial, usadas em conjunto, criaram importantes ferramentas estratégicas, os ERP's, que as empresas passaram a utilizar. Estas novas ferramentas passaram a equipar as empresas com as capacidades necessárias para integrar e sincronizar processos isolados, afim de integrar o processo de negócio, de forma a tornarem-se mais competitivas no mercado actual.

ERP é um conceito genérico que pretende identificar o conjunto de actividades executadas por um *software* modular e tem por objectivo primário, o auxílio dos processos de gestão de uma empresa nas mais importantes fases de seu negócio [14].

De uma forma abrangente e integrada, estas actividades incluem, por exemplo, o desenvolvimento de produto; a compra de matéria-prima e componentes; a interacção com fornecedores e clientes; o acompanhamento de ordens de produção; o serviço a clientes; a gestão de stocks; a gestão contabilística e financeira; a gestão de recursos humanos; a gestão da qualidade; a gestão de projectos; entre outros [14]. Na figura 2.6 está ilustrada uma estrutura típica dos sistemas ERP.

Estes produtos visam essencialmente eliminar a redundância de operações e a burocracia, por meio da automatização de processos. Assim, os módulos que compõem o ERP possibilitam, desenvolver e gerir o negócio de forma integrada em tempo real. Além disso, as informações tornam-se mais consistentes, possibilitando a tomada de decisão com base em dados que reflectem a realidade da empresa num dado momento [14]. Os ERP's ajudam as empresas a interligar todos os seus recursos, gerindo-os da melhor forma possível

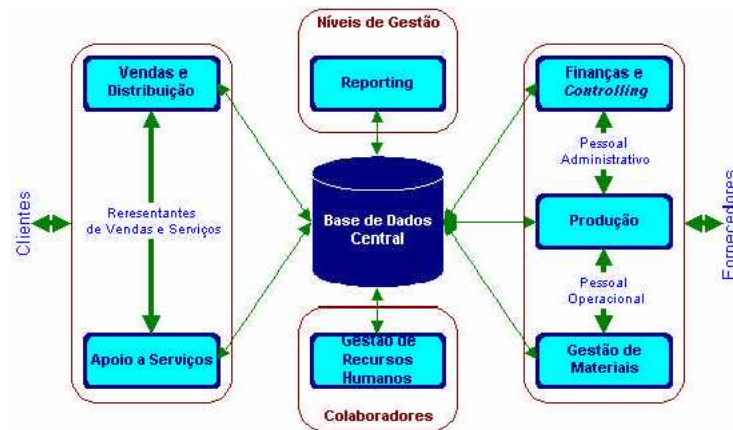


Figura 2.6: Estrutura de um ERP [14]

em tempo real. Estes permitem também às empresas alcançarem índices de performance elevados, à medida que linearizam e otimizam todos os seus recursos.

Na sua concepção fundamental, o ERP é um sistema aplicativo que define uma infra-estrutura básica (*backbone*) para toda a empresa. Ele integra processos de gestão e de negócios, proporcionando uma visão global da organização [35].

2.3.1 Potencialidades dos Sistemas ERP

Segundo Sousa [42], O ERP é um sistema que possui inúmeras potencialidades, tais como:

- Facilita a existência de um sistema de informação integrado de todas as áreas funcionais de uma empresa;
- Executa as tarefas críticas de uma empresa, aumentando a qualidade dos serviços a clientes e melhorando a imagem da empresa;
- Possibilita a troca de informação em ambientes distribuídos;
- Possibilita a integração de informação dos vários departamentos, escritórios, fábricas de uma empresa, bem como das várias empresas pertencentes a um grupo financeiro;

- Constitui a melhor solução para uma eficiente gestão de projectos;
- Resolve muitos dos problemas comuns das empresas: gestão de *stocks*, serviços a clientes, gestão financeira, controlo de qualidade, entre outros;
- Não se destina exclusivamente às necessidades da empresa uma vez que fornece oportunidades de um melhoramento contínuo, face às evoluções de mercado, entre outros.

2.3.2 Implementação de um Sistema ERP

O sistemas ERP's são parametrizados de formas diferentes, de acordo com cada tipo de empresa e o ramo em que esta está inserida. Para flexibilizar este facto, os sistemas ERP foram desenvolvidos para apresentarem uma solução genérica, de forma a que esta possa ser parametrizada em certo grau, tendo em atenção, no entanto, que esta parametrização acaba por ser um compromisso entre os requisitos da empresa e as funcionalidades oferecidas pelo sistema. Desta forma, é necessário que alguns processos de negócio das empresas obriguem a uma redefinição para que os seus requisitos se enquadrem nas funcionalidades oferecidas pelo sistema [42].

Um *software* ERP é composto por vários módulos que poderão ser instalados separadamente. A primeira medida a tomar no processo de instalação é a selecção dos módulos que serão instalados numa primeira fase. Contudo, é possível que posteriormente sejam integrados novos módulos. Seguidamente, cada módulo instalado é parametrizado para que se adeque da melhor forma ao processo de negócio da empresa. No entanto, mesmo com esta parametrização, a solução final poderá não corresponder aos requisitos da empresa, e, nesses casos, as empresas necessitam de utilizar outros sistemas complementares ou abandonar por completo a instalação deste tipo de sistema e optar por processos genéricos. Por este motivo, a decisão de implementar um sistema ERP só deverá ser tomada no seguimento de uma análise detalhada dos processos da empresa e das funcionalidades oferecidas pelas soluções do sistema [42].

Em Martins [29] apresenta-se e discute-se uma lista de dez factores considerados

críticos para o sucesso de uma implementação de ERP, sendo estes:

- Obter a participação activa da gerência (*Commitment*);
- Implementar a gestão de mudanças tentando reduzir a insegurança dos utilizadores pouco informados;
- Identificar os administradores, que são indispensáveis nos seus respectivos departamentos;
- Escolher com segurança um profissional experiente e respeitado, para gerir o projecto, de modo a descaracterizar o ERP como um sistema da área de informática, e sim como um redesenho do modelo de gestão;
- Planear e realizar "ensaios/testes";
- Definir claramente os diversos papéis na implementação do sistema, através da união de conhecimentos e esforços para o alcance do sucesso;
- Adaptar o sistema à empresa e vice-versa, reflectindo sobre a realidade actual da empresa ou a utilização das melhores práticas (*best-practices*);
- Escolher a consultoria adequada (*know-how*);
- Garantir a qualidade (*quality assurance*);
- Simplificar em todos os sentidos: na definição de modelos, no desenho da solução e na própria implementação do sistema.

2.3.3 Pontos Críticos do ERP

Alguns pontos e características importantes dos sistemas ERP devem ser cuidadosamente analisados no momento da aquisição e implantação dos mesmos. De acordo com Martins [29], existe um conjunto de factores críticos a ter em conta num sistema ERP:

- São aplicações comerciais desenvolvidas a partir de modelos-padrões de processos, que não são específicos para uma determinada necessidade, mas sim genéricos, podendo a empresa adequar-se ou não a eles;
- Integram todas as áreas da empresa, sendo uma vantagem a utilização destas ferramentas. A empresa obtém integridade e confiabilidade nas informações adquiridas através do sistema, pois a entrada de um dado ocorre uma única vez dentro do mesmo, que a partir de então passa a actualizar automaticamente todos os módulos necessários;
- Permitem a adequação das funcionalidades existentes no sistema às da empresa, através do processo de parametrização. Este processo consiste na definição de diversos valores que são introduzidos no sistema, com o intuito de dimensionar o perfil da empresa e o comportamento do sistema;
- Eles possibilitam a personalização de determinados processos de *software* que não se adaptam à empresa, mesmo usando a parametrização. A personalização é a adaptação do sistema às necessidades específicas da empresa, sendo necessário intervir com programas ou rotinas que se integram ao ERP. Muitas actividades da empresa não são contempladas pelo sistema, não bastando apenas configurá-lo através de parâmetros. Esta etapa nem sempre é realizada pela empresa que desenvolve o ERP, e muitas vezes uma consultoria homologada e conhecedora da solução é contratada para este fim;
- Possuem custos elevados, destacando-se os custos de *hardware* e infra-estrutura computacional, de aquisição da licença de uso do ERP, e de consultoria para a implantação. Um sistema ERP apresenta muitas complexidades, sendo que a sua implantação deverá ser realizada por profissionais que conheçam, não somente o negócio da empresa, como também a solução escolhida. Geralmente, as empresas optam por contratar consultores especializados no produto escolhido. Também os utilizadores dos vários departamentos deverão passar por um período no qual os esforços serão duplicados, uma vez que o trabalho deverá ser realizado paralelamente no sistema antigo (mesmo que manual) e no novo.

- Os fornecedores de sistemas ERP disponibilizam periodicamente versões actualizadas (*upgrades*) que congregam melhorias, correcções de problemas e erros do sistema. Este processo de actualização deve ser flexível e permitir a adequação da nova versão com possíveis personalizações efectuadas no produto.
- Os sistemas ERP forçam, na maioria das vezes, alterações nos processos produtivos e administrativos, pois é necessária tanto a adaptação do sistema aos processos da empresa, como a adaptação da empresa a determinados processos do sistema. Estas alterações são complexas e podem causar, no início, uma série de inconvenientes até que todos estejam adaptados à nova realidade. É importante ressaltar também que estas alterações de processos devem estar em conformidade com as estratégias da empresa e os seus objectivos de longo prazo, merecendo, portanto, grandes cuidados em sua implementação.
- O ERP tem impacto sobre os recursos humanos da empresa, pois as pessoas terão que se preocupar com o processo como um todo e não apenas com a sua actividade específica. Devido à integração do sistema, um problema de uma área poderá se reflectir-se rapidamente noutros departamentos, existindo o risco de chegar a afectar toda a empresa. O perfil dos profissionais muitas vezes será alterado, uma vez que se exigirá multi-disciplinariedade e conhecimentos que nem sempre os actuais funcionários possuem. A empresa deverá optar por reciclar os seus profissionais, ou às vezes substituí-los. Esta última alternativa é reforçada também pelo facto de que a partir da automação e, mais do que isso, da integração entre os processos, muitas actividades que eram realizadas manualmente, no sistema anterior, não serão mais necessárias. Muitas vezes, pode ocorrer resistência interna à adopção do ERP, devido à desconfiança de perda de emprego ou de poder, uma vez que haverá maior partilha da informação.
- Os sistemas ERP apresentam dificuldades no cumprimento de prazos de instalação e orçamentos, devido a: resistência por parte das pessoas; rotatividade dos funcionários que foram treinados no novo sistema ou que dominam o negócio da empresa; qualidade dos recursos humanos internos e da equipa de consultoria contratada; limitações inerentes ao próprio produto ERP escolhido e dificuldade de

integrar o ERP com outros sistemas existentes dentro da empresa. Todos estes factores não podem ser correctamente previstos com antecedência no momento de elaboração dos cronogramas e orçamentos, e, por mais que se possa inserir margens de segurança, eles podem comprometer a credibilidade do projecto.

2.4 Tecnologias

A gestão e publicação da informação dos modelos de documentos é efectuada baseada nas tecnologias *standard* XML e XSL. O XML permite armazenar a informação e o XSL permite fazer a apresentação do conteúdo XML.

A sub-secção 2.4.1 apresenta o XML e a sub-secção 2.4.2 apresenta o XSL.

2.4.1 XML

Esta secção não pretende descrever exaustivamente a norma XML e todas as suas componentes, mas sim dar uma visão global do seu propósito e da sua estruturação, bem como permitir entender o seu enquadramento no tema da integração.

A linguagem XML é um subconjunto de outro *standard* lançado em 1986, o SGML da norma ISO 8879. O SGML foi o primeiro *standard* de linguagens de anotação de texto e o seu objectivo era normalizar a produção de documentos. Para isso, dispunha e dispõe de uma série de características invulgaes das quais se destacam:

- Separação completa do conteúdo e da aparência visual - o SGML preocupa-se com a estrutura da informação e não com a forma com que esta vai ser apresentada ao utilizador;
- É completamente independente de plataformas de *hardware* e *software* - um documento SGML pode ser transportado entre sistemas ou aplicações.

Pode-se considerar o XML uma *meta-markup language*, uma linguagem de anotação descritiva extensível, tendo todas as características que tornam desejável este tipo

de linguagens: independência relativamente às plataformas de *software* e de *hardware* utilizadas, longevidade, custos de manutenção reduzidos e fácil utilização. É uma linguagem que disponibiliza um formato para a descrição de dados estruturados. O XML é uma metalinguagem que em qualquer momento permite acrescentar novos elementos à linguagem, o que torna esta linguagem, aberta e sem limitações do ponto de vista de extensibilidade [20].

Os documentos XML tem uma estrutura lógica em forma de árvore (ver figura 2.7). Existe sempre um elemento que contém todos os outros (o elemento raiz correspondente à raiz da árvore estrutural). A estrutura em árvore do documento é utilizada pela maior parte das aplicações para ler ou processar o documento.

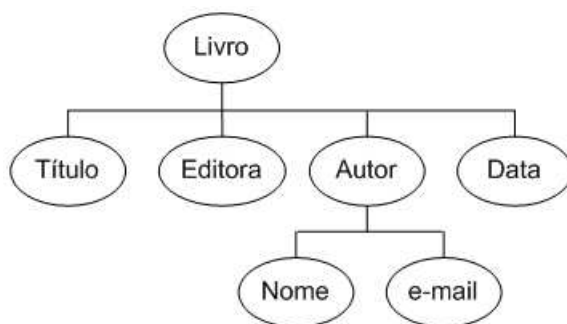


Figura 2.7: Estrutura em árvore dum documento XML

À semelhança do HTML o XML permite a definição de *tags* ou delimitadores, que caracterizam os dados e a sua formatação, mas de forma ilimitada. É possível criar um conjunto de *tags* ou marcas específicas para obter uma formatação para um determinado conjunto de informação. Cada elemento, ou conjuntos de elementos, está relacionado e estruturado numa lógica de árvore ou hierarquia com dependências e ramificações. Uma folha da árvore será um elemento que pode conter dados como o título de um livro, o código de um produto ou qualquer outro tipo de dados. Com o XML pode-se criar uma estrutura para um documento de texto, um registo de base de dados estruturado, um objecto com métodos e dados, como são os objectos Java, um registo de dados resultantes de uma pesquisa, a apresentação gráfica de uma aplicação, entre outros.

Os elementos XML são formados por *tags* de abertura <elemento>, sendo o mesmo

```
<? XML version="1.0" encoding="ISO-8859-1" ?>
<livros>
  <livro id=1>
    <titulo>Titulo do livro 1</titulo>
    <editora>Editora do livro 1</editora >
    <autor>
      <nome>Nome do autor 1</nome >
      <email>e-mail do autor 1</email>
    </autor>
    <data>Data publicação 1</data>
  </livro>
  <livro id=2>
    <titulo>Titulo do livro 2</titulo>
    <editora>Editora do livro 2</editora >
    <autor>
      <nome>Nome do autor 2</nome >
      <email>e-mail do autor 2</email>
    </autor>
    <data>Data publicação 2</data>
  </livro>
  ...
</livros>
```

Figura 2.8: Estrutura de um documento XML

para *tags* de fecho `</elemento>`, como pode ser observado na figura 2.8. Por exemplo, a definição do elemento título pode ser representada da seguinte forma: `<titulo>Título</titulo>`. Um documento XML estrutura-se com pelo menos três partes: uma declaração, o DTD e a instância. A declaração funciona como um cabeçalho que indica a versão de XML utilizada, o grupo de caracteres possíveis (codificação) ou outras características. Um DTD caracteriza-se por um conjunto de declarações que especifica o tipo de estrutura do documento em XML (figura 2.9). Contudo, os DTDs são opcionais e os dados enviados com um DTD são conhecidos como dados XML válidos. Um processador de documentos pode analisar os dados que chegam, analisando as regras contidas no DTD para verificar se os dados foram estruturados correctamente. Os dados enviados sem DTD, são conhecidos como dados bem formatados, e, nesse caso, o documento pode ser usado para implicitamente se auto-descrever. A instância é o documento propriamente dito, contendo a informação e as anotações. Na própria instância poderá fazer-se referência ao DTD, caso este não esteja presente directamente no documento XML. Também existe a abordagem XSD que se posiciona como uma alternativa mais completa e flexível do que o DTD. O XSD foi proposto inicialmente pela Microsoft e passou a ser uma recomendação oficial do W3C em 2001.

```
<? XML version="1.0" encoding="ISO-8859-1" ?>

<!ELEMENT note (to,from,heading,body)>
<!ELEMENT to      (#PCDATA)>
<!ELEMENT from    (#PCDATA)>
<!ELEMENT heading (#PCDATA)>
<!ELEMENT body    (#PCDATA)>

<note>
  <to>. . .</to>
  <from>. . .</from>
  <heading>. . .</heading>
  <body>. . .</body>
</note>
```

Figura 2.9: *Document Type Declaration*

Este formato permite partilhar os dados com os quais se trabalha a todos os níveis, por todas as aplicações e suportes. Sendo assim, o XML¹ tem um papel importantíssimo no mundo actual, que tende à globalização e à compatibilidade entre os sistemas distintos, já que é a tecnologia que permitirá partilhar a informação de uma forma segura, confiável e fácil (ver figura 2.10).

¹Para mais informações sobre XML consultar a página <http://www.w3schools.com/xml/>

Vantagens do XML

Segundo Afonso [4] as principais vantagens da linguagem XML são:

- O XML é extensível. A possibilidade de criar *tags* de um modo arbitrário (respeitando sempre as regras de aninhamento), permite adaptar a estrutura de um documento XML a praticamente qualquer situação específica;
- Os documentos XML são auto-descritivos. São relativamente fáceis de interpretar, manipular e interrogar. Esta característica pode também revolucionar o modo como as pesquisas são efectuadas na *web*, permitindo o aparecimento de motores de pesquisa que realizem as pesquisas tendo em conta o significado (contexto) dos dados, em vez de se basearem unicamente na associação de palavras-chave;
- Apesar da sua simplicidade, o XML permite criar estruturas bastante complexas (árvores ou grafos de profundidade arbitrária e, eventualmente, cíclicos e recursivos);
- O XML é extremamente flexível, possibilitando a representação, quer de dados estruturados, quer de dados semi-estruturados;
- Recorrendo à definição de um esquema, é possível efectuar a validação de documentos. Esta característica revela-se de extrema importância para aplicações em que a verificação estrutural de dados seja vital, como é o caso das bases de dados convencionais;
- O conteúdo de um documento XML pode ser facilmente manipulado pelas aplicações de *software* (recorrendo às API's existentes), o que torna possível atingir níveis de automação bastante elevados;
- Uma vez que o XML tem uma natureza meta-linguística, as organizações podem utilizá-la para desenvolver padrões específicos (novas linguagens baseadas em XML, como, por exemplo, a MathML²), definindo esquemas comuns, de modo a trocarem eficientemente dados entre si. Estes esquemas podem ser disponibilizados

²A sua especificação (versão 3.0) está disponível em <http://www.w3.org/TR/MathML3/>

publicamente na *web*. O objectivo é utilizar o XML para a troca de dados entre os sistemas de informação organizacionais;

- O XML é um padrão aberto. Os documentos XML são independentes das aplicações, dos sistemas operativos, entre outros. Esta característica pode vir a revolucionar a integração de sistemas heterogéneos;
- Uma vez que o conteúdo de um documento XML está separado da sua apresentação, é possível obter múltiplas perspectivas sobre um mesmo documento XML recorrendo à XSL (ver sub-secção 2.4.2).

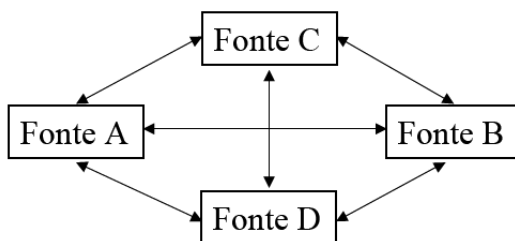


Figura 2.10: Troca de dados entre sistemas heterogéneos [4]

2.4.2 XSL

O XML pode ainda ser complementado com o XSL para controlar a apresentação dos dados. O XML permite estruturar e representar informação de uma forma simples e independente da sua origem ou do seu destino. Um documento que contém informação formatada pelo XML pode não ser suficientemente flexível para ser manipulado, sobretudo quando não se conhece à partida a sua estruturação. Para facilitar essa interpretação e visualização do conteúdo em XML, surge outra norma do W3C: o XSL. O XSL permitirá substituir as *tags* de estruturação da informação por marcas de formatação ou transformação semelhante ao que acontece com o HTML.

O XSL é uma linguagem especialmente concebida para controlar a apresentação dos dados armazenados em ficheiros XML. Um documento XSL é composto por instruções

que percorrem e seleccionam os nós de um documento XML e é capaz de transformá-lo em outro documento, ou num outro formato de representação. Logo, os documentos XML podem ser representados de tantas formas quantos sejam os documentos XSL existentes, sem que para isso tenham que ser modificados os documento XML.

A ideia fundamental da utilização de XML e XSL é separar conteúdo de forma de apresentação. O conteúdo em XML possui marcadores muito simples, que podem ser lidos, tanto por humanos como por máquinas. O uso de XSL permite especificar como se quer que o texto em XML seja apresentado ao utilizador.

O XSL foi criado pela necessidade de uma linguagem de apresentação dos dados XML. O XSL consiste em três partes principais: XSLT, XPath e XSL-FO.

- XSLT (*XSL Transformations*): linguagem para descrever como transformar um documento XML em outro documento;
- XPath (*XML Path Language*): linguagem utilizada pelo XSLT para aceder ou referenciar partes específicas de um documento XML;
- XSL-FO (*XSL Formatting Objects*): um vocabulário XML para especificação da formatação da semântica.

Na sua primeira abordagem, o W3C definiu o XSL com ambas as partes de formatação e de transformação do XML, e mais tarde decidiu separar nas três partes acima referidas.

XSLT

O XSLT era originalmente parte do XSL. Na realidade, continua tecnicamente a fazer parte do XSL, segundo a especificação, a descreve como uma linguagem constituída por duas partes: uma linguagem para transformar documentos XML e um vocabulário para descrever como formatar conteúdo de documentos. O W3C verificou que poderia converter documentos XML em qualquer outro tipo de documento. Chamaram a esta

linguagem XSLT, e separaram-na na sua própria especificação, embora também seja considerada parte da especificação do XSL [13].

XSLT é uma linguagem para transformar a estrutura de um documento XML [27]. De acordo com Fung [16], a finalidade original do XSLT era transformar os documentos XML em documentos XSL. Porém, como definida agora, o XSLT pode também transformar documentos XML em páginas HTML, documentos de texto comum e documentos de texto com estruturas não definidas no XML. Ou seja, com os dados de um documento XML é possível criar outro documento com apresentação semelhante à página HTML ou outro formato de texto.

Na figura 2.11, é apresentado o exemplo de código de XSLT, cujo documento resultante terá os seus dados apresentados como HTML. Neste exemplo podem ser identificadas as *tags* HTML para formatar a apresentação. Na figura 2.12, está ilustrado o resultado do documento gerado a partir da folha de estilo XSLT, aplicada ao documento XML apresentado na figura 2.8.

XPath

Em XPath, um documento XML é visualizado, conceptualmente, como uma árvore, onde cada parte do documento é representada como um nó. Existem sete tipos de nós: raiz, elemento, atributo, texto, comentário, instrução de processamento e espaço de nomes.

Para exemplificar os vários tipos de nós utiliza-se o documento XML apresentado na figura 2.13.

Segundo Harold [19], uma árvore XPath possui um nó raiz que contém todos os outros nós da árvore. O nó raiz e os nós elementos contém uma lista ordenada de todos os nós filhos. Cada nó, excepto o nó raiz, tem um nó pai e todos os nós pai possuem um nó filho ou descendente. Os nós que podem ter nós filhos são os seguintes: elemento, comentário, texto e instrução de processamento.

Todos os elementos no documento XML original são representados por um nó (nó

```

<?xml version="1.0" encoding="ISO-8859-1"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
<xsl:output method="html" encoding="ISO-8859-1"/>
<xsl:template match="/">
<html><head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1"/>
<title>Livros</title>
<style type="text/css">
<![CDATA[ body{ color: #000000;
background: #ffffff;
font-family: Helvetica, Arial }
th{ text-align: left;
font-family: Times;
color: #ffffff;
background: #ff0000;
vertical-align: bottom; }

tr { vertical-align: top; }

td { font-size: 80%;
color: #000000;
background: #ffffff;
vertical-align: top; }
]]>
</style>
</head>
<body>
<!-- Cabeçalho -->
<center> <b>Livros</b> <hr/> </center>
<center>
<table cellSpacing="0" width="600" border="0" cellPadding="4">
<tr bgColor="#00a000">
<th width="150" style="background: #aaaaaa">
<b>Titulo</b>
</th>
<th width="150" style="background: #aaaaaa">
<b>Editora</b>
</th>
<th width="175" style="background: #aaaaaa">
<b>Data</b>
</th>
<th width="175" style="background: #aaaaaa">
<b>Autor</b>
</th>
</tr>
<xsl:apply-templates select="livros/livro"/>
</table>
</center>
</body>
</html>

</xsl:template>
<xsl:template match="livro"> <!-- Inicio Livro -->
<tr>
<td width="150" vAlign="top"> <!-- Titulo -->
<b> <xsl:apply-templates select="titulo"/> </b>
</td>
<td width="150" vAlign="top"> <!-- Editora -->
<xsl:apply-templates select="editora"/>
</td>
<td width="175" vAlign="top"> <!-- Data -->
<xsl:apply-templates select="data"/>
</td>
<td width="175" vAlign="top" colSpan="3"> <!-- Autor -->
<font size="-2">
Nome: <xsl:apply-templates select="autor/nome"/> <br/>
eMail: <xsl:apply-templates select="autor/email"/>
</font>
</td>
</tr>
<tr>
<td width="100%" colSpan="6"> <hr/> </td>
</tr>
</xsl:template> <!-- Fim livro -->
</xsl:stylesheet>

```

Figura 2.11: Exemplo de um XSLT

Livros			
Titulo	Editora	Data	Autor
Titulo do livro 1	Editora do livro 1	Data publicação 1	Nome: Nome do autor 1 eMail: e-mail do autor 1
Titulo do livro 2	Editora do livro 2	Data publicação 2	Nome: Nome do autor 2 eMail: e-mail do autor 2

Figura 2.12: Transformação de um documento XML através de XSLT para HTML

```
<? XML version="1.0" encoding="ISO-8859-1" ?>
<livro id="engenharia">
  <titulo>Titulo do livro</ titulo>
  <editora>Editora do livro</ editora >
  <autor>
    <nome>Nome do autor</nome >
    <email>e-mail do autor</email>
  </autor>
  <data>Data publicação</data>
</livro>
```

Figura 2.13: Nós de um documento XML

elemento). A árvore terá nós deste tipo para o elemento livro, titulo, editora, autor, data entre outros. Os filhos de um nó elemento podem ser nós texto, nós elemento, nós comentário e nós instrução de processamento que ocorrem nesse elemento no documento original. O valor textual de um nós elemento é a concatenação do texto deste nós e de todos os seus filhos, pela ordem em que estes aparecem no documento. Como já foi mencionado, todas as referências a entidades foram expandidas; não se pode manipular entidades gerais ou carácter em XPath.

Os nós texto são os mais simples, apenas contêm o texto do elemento correspondente. Se no documento original existirem referências a entidades, estas serão resolvidas antes de o nó ser criado, sendo que um nó texto contém apenas texto puro. Além disso, um nó texto contém o máximo de texto possível, o que faz com que um nó destes não tenha irmãos do seu tipo na árvore documental.

Um nó atributo tem sempre um pai que é um nó elemento. No exemplo da figura 2.13, o nó elemento correspondente ao elemento livro é pai de um nó atributo com nome

tipo e valor engenharia. Este tipo de nós tem, no entanto, algumas funcionalidades distintas dos outros tipos de nós [20]:

- Apesar de um nó elemento ser o pai dos seus nós atributos, estes não são filhos do seu pai. Os filhos de um nó elemento são os elementos texto, elemento, comentário e instrução de processamento contidos no elemento correspondente do documento original. Se se quiser seleccionar os nós atributo, é necessário indicar explicitamente o mesmo.
- Em circunstâncias normais, o processador de XPath cria um nó atributo para todos os atributos instanciados no documento original e para os atributos que tenham um valor por omissão declarado no DTD (se o processador tiver capacidade para analisar um DTD).

Um nó comentário é também muito simples, e contém apenas texto. O texto de um nó comentário contém todo o conteúdo do comentário original, excepto as marcas `<!--` e `-->`.

Um nó instrução de processamento tem duas partes: um nome e um valor textual. O valor textual é tudo o que aparece a seguir ao nome, excepto a marca de fecho `?>`.

Os nós *namespaces* raramente são utilizados nas folhas de estilo XSL. Estes nós existem principalmente para benefício do processador que tem de diferenciar elementos pertencentes a diferentes *namespaces*. As declarações de *namespaces* nas instâncias XML, apesar de corresponderem tecnicamente a atributos, são tratadas em XPath como nós do tipo *namespace* (uma declaração como: `xmlns:autor="http://www.autor.pt"`, seria armazenada num nó do tipo *namespace*).

Para percorrer o caminho da árvore, é utilizada a linguagem XPath, juntamente com XSLT para que os valores de um documento XML possam ser acedidos. XPath define uma biblioteca de funções padrão, sendo o elemento principal em XSLT [43], pois com ela é possível percorrer todos os nós da árvore podendo-se adquirir quaisquer valores nela contida. Logo, é possível obter valores de filhos, irmãos ou qualquer parente de um nó em questão.

Para localizar uma parte de um documento na estrutura de árvore definida em XPath, é usado um caminho de localização. Um caminho de localização é uma expressão que especifica como navegar numa árvore XPath de um nó para outro. Um caminho de localização é composto por passos de localização, sendo que cada um é composto por um eixo, um nó teste e um nó predicado, opcional [10].

XML permite a descrição de dados de uma forma flexível e eficiente com a utilização de marcadores descritivos. No entanto, ele não oferece meios para localizar partes de dados dentro de um documento. Com a utilização de XPath pode-se localizar partes específicas de um documento XML de forma eficiente [10]. XPath é uma linguagem de expressão que permite o processamento de valores de acordo com um modelo de dados [17]. Este modelo contém uma árvore de representação de um documentos XML, bem como valores atômicos, tais como inteiros, strings e booleanos. O resultado de uma expressão XPath pode ser uma selecção de nós de um documento ou um valor atómico.

XSL-FO

XSL-FO é uma linguagem de anotação XML para formatação de documentos. É uma linguagem que descreve como as páginas serão apresentados aos leitores, ou seja, como os utilizadores irão visualizar a informação. Uma folha de estilo utiliza a linguagem XSLT para transformar um documento XML com um vocabulário semântico, num novo documento XML que usa o vocabulário de apresentação XSL-FO.

Segundo Pawson [34], o XSL-FO efectua um exame à entrada de XML e entrega uma cópia, ou seja, um documento XML aplicado ao código FO, que passa por um processador que formata este documento transformando-o num outro arquivo, geralmente no formato PDF ou postScript1. Entre a entrada e a saída da cópia, existe o intermediário *namespace* de FO, que reconhece elementos e atributos que podem ser utilizados na criação do código.

O XSL-FO é tipicamente utilizado quando o resultado da transformação é uma mídia impressa, como livros e revistas. Um documento XML é transformado num documento XSL que marca os dados usando objectos de formatação. Este documento XSL

pode ser então transformado noutros formatos, por exemplo, PDF ou TEX.

Na figura 2.14 pode observar-se o processo transformação do documento XML para outro tipo de documento.



Figura 2.14: Transformação de documentos XML [6]

Pires [36] afirma que, isoladamente, XSL-FO não é uma folha de estilo. É um formato final, com uma estrutura que visa a apresentação. Geralmente, cria-se um XSLT que utiliza código FO, e a partir de uma fonte XML gera um PDF. Para que uma folha de estilo XSLT possa utilizar elementos de FO, deve ser denominado o seu *namespace*. Para percorrer por inteiro e transformar o documento XML é utilizada a linguagem XPath, visto que é parte essencial do XSLT.

Estrutura XSL-FO

Nesta secção é demonstrado a estrutura do código XSL-FO. Em analogia com o HTML que possui o nó raiz <html> e é dividido em <head> e <body>, no XSL-FO o nó raiz é <fo:root> dividindo-se em <fo:layout-master-set> e <fo:page-sequence>. Na figura 2.15 está representada a estrutura do código.

- O elemento <fo:layout-master-set> define os layouts disponíveis:
 - <fo:simple-page-master>: Define as sub-divisões das páginas e sua forma (ver figura 2.16:
 - * <fo:region-body>: A parte central da página (obrigatória);
 - * <fo:region-before>: Cabeçalho;

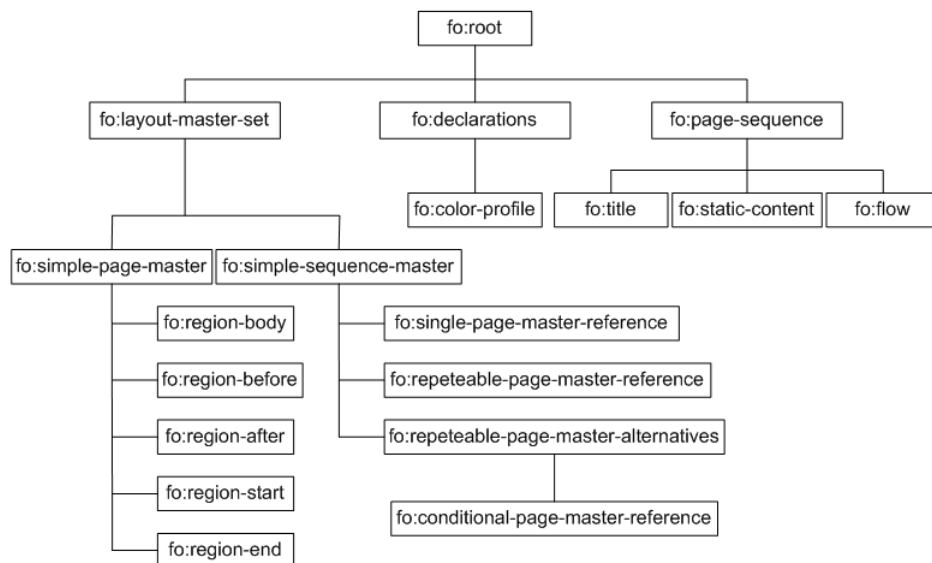


Figura 2.15: Estrutura do XSL-FO

- * `<fo:region-after>`: Rodapé;
- * `<fo:region-start>`: Coluna a esquerda;
- * `<fo:region-end>`: Coluna a direita.
- `<fo:page-sequence-master>`: Define como os padrões acima se relacionam:
 - * `<fo:single-page-master-reference>`: Modelo de uma única página;
 - * `<fo:repeteable-page-master-reference>`: Modelo único que pode se repetir;
 - * `<fo:repeteable-page-master-alternatives>`: Modelo que escolhe um dentre diversos layouts.
- `<fo:page-sequence>`:
 - `<fo:static-content>`: Conteúdo das partes estáticas do documento;
 - `<fo:flow>`: Corpo do documento.

Na figura: 2.17 é apresentado um exemplo de código XSL-FO, e na figura 2.18 pode ser visualizado um documento PDF onde é apresentado o resultado da transformação da informação XML aplicada ao XSL-FO.

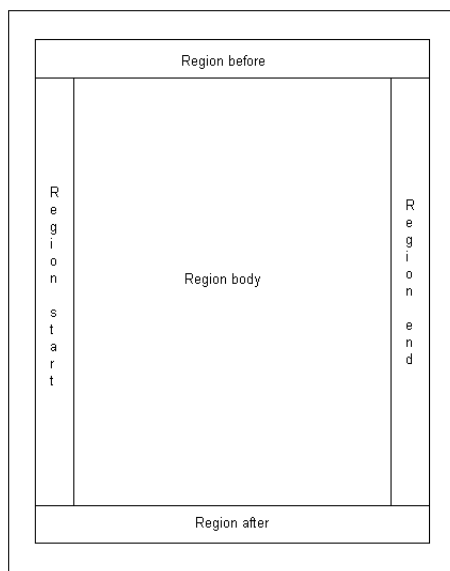


Figura 2.16: Sub-divisões da página

```

<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform" xmlns:fo="http://www.w3.org/1999/XSL/Format">
<xsl:output method="xml" encoding="UTF-8" indent="yes"/>

<xsl:template match="/">
  <fo:root>
    <fo:layout-master-set>
      <!-- Corpo da página -->
      <fo:simple-page-master master-name="default-master" page-height="297mm" page-width="210mm" margin-top="0.25cm" margin-bottom="0.25cm">
        <!-- Zona central da página -->
        <fo:region-body column-count="2" margin-top="5.0mm" margin-bottom="5.0mm"/>
        <!-- Cabeçalho -->
        <fo:region-before extent="5mm"/>
        <!-- Rodapé -->
        <fo:region-after extent="5mm"/>
      </fo:simple-page-master>
    </fo:layout-master-set>
    <!-- Corpo da página -->
    <fo:page-sequence master-reference="default-master">
      <!-- Conteúdo do cabeçalho -->
      <fo:static-content flow-name="xsl-region-before">
        <fo:block font-size="8.0pt" font-family="serif" padding-after="2.0pt" space-before="4.0pt" text-align="center" border-bottom-style="solid" border-bottom-width="1.0pt">
          <xsl:text>Livros</xsl:text>
        </fo:block>
      </fo:static-content>
      <!-- Conteúdo do rodapé -->
      <fo:static-content flow-name="xsl-region-after">
        <fo:block font-size="8.0pt" font-family="sans-serif" padding-after="2.0pt" space-before="4.0pt" text-align="center" border-top-style="solid" border-top-width="1.0pt">
          <xsl:text>Página </xsl:text><fo:page-number/>
        </fo:block>
      </fo:static-content>
      <!-- Conteúdo do corpo da página -->
      <fo:flow flow-name="xsl-region-body">
        <fo:block font-size="10.0pt" font-family="serif">
          <xsl:apply-templates select="/result/videos/video">
            <xsl:sort select="*[contains(name(), 'title')]" />
          </xsl:apply-templates>
        </fo:block>
      </fo:flow>
    </fo:page-sequence>
  </fo:root>
</xsl:template>

<!-- Formato das entradas (imagem) livros.xml -->
<xsl:template match="video">
  <xsl:call-template name="entry">
    <xsl:with-param name="image" select="concat('url(', $examples-dir, '/images/livros/', @id, '.gif', ')')"/>
  </xsl:call-template>
</xsl:template>

<xsl:template name="entry">
  <xsl:param name="image" select="''"/>
  <!-- Apresentação do livro -->
  <fo:block space-before="4.0pt" space-after="4.0pt" start-indent="0.25cm" end-indent="0.25cm">
    <!-- Título -->
    <fo:block font-family="sans-serif" font-size="12.0pt" space-after="2.0pt">
      <xsl:value-of select="titulo"/>
    </fo:block>
    <!-- Nome do autor -->
    <fo:block font-family="sans-serif" space-after="2.0pt">
      <xsl:text>by </xsl:text>
      <xsl:call-template name="listar">
        <xsl:with-param name="list" select="autor/nome"/>
      </xsl:call-template>
    </fo:block>
    <!-- email do autor -->
    <fo:block font-family="sans-serif" font-size="8.0pt" space-after="2.0pt">
      <xsl:text>email: </xsl:text>
      <xsl:call-template name="listar">
        <xsl:with-param name="list" select="autor/email"/>
      </xsl:call-template>
    </fo:block>
    <!-- Imagem (opcional) -->
    <xsl:if test="$image">
      <fo:block text-align="center">
        <fo:external-graphic src="{ $image }"/>
      </fo:block>
    </xsl:if>
    <!-- Editora -->
    <fo:block font-family="sans-serif" font-size="8.0pt" space-after="2.0pt">
      <xsl:text>Editora: </xsl:text>
      <xsl:call-template name="listar">
        <xsl:with-param name="list" select="editora"/>
      </xsl:call-template>
    </fo:block>
    <!-- Data -->
    <fo:block font-family="sans-serif" font-size="8.0pt" space-after="2.0pt">
      <xsl:text>Data: </xsl:text>
      <xsl:call-template name="listar">
        <xsl:with-param name="list" select="data"/>
      </xsl:call-template>
    </fo:block>
    <fo:block border-top-style="solid" border-bottom-width="1.0pt"/>
  </fo:block>
</xsl:template>

<!-- Listar livros -->
<xsl:template name="listar">
  <xsl:param name="list"/>
  <xsl:value-of select="$list[1]"/>
</xsl:template>
</xsl:stylesheet>

```

Figura 2.17: Exemplo do código XSL-FO

Livros	
Titulo do livro 1 by Nome do autor 1 email: e-mail do autor 1 Capa ? Livro Editora: Editora do livro 1 Data: Data de publicação 1	Capa ? Livro Editora: Editora do livro 4 Data: Data de publicação 4
Titulo do livro 2 by Nome do autor 2 email: e-mail do autor 2 Capa ? Livro Editora: Editora do livro 2 Data: Data de publicação 2	Capa ? Livro Editora: Editora do livro 5 Data: Data de publicação 5
Titulo do livro 3 by Nome do autor 3 email: e-mail do autor 3 Capa ? Livro Editora: Editora do livro 3 Data: Data de publicação 3	Capa ? Livro Editora: Editora do livro 6 Data: Data de publicação 6
Titulo do livro 4 by Nome do autor 4 email: e-mail do autor 4	Titulo do livro 5 by Nome do autor 5 email: e-mail do autor 5
	Titulo do livro 6 by Nome do autor 6 email: e-mail do autor 6
	Titulo do livro 7 by Nome do autor 7 email: e-mail do autor 7

Página 1

Figura 2.18: Transformação de um documento XML para PDF

2.4.3 Frameworks JavaScript

No desenvolvimento do *software*, uma *framework* é uma estrutura de suporte definida, em que um outro projecto de *software* pode ser organizado e desenvolvido. Uma *framework* pode incluir programas de suporte, bibliotecas de código, linguagens de *script* e outros *softwares* para apoiar o desenvolvimento e o agrupamento de diferentes componentes de um projecto de *software* [2].

Framework é uma solução reutilizável de *software*, assim como o é uma biblioteca. No entanto, a *framework*, num certo ponto, assume o controlo do fluxo de execução, sendo o programador responsável por escrever apenas os seus pontos de extensão enquanto a biblioteca constitui apenas um conjunto de funcionalidades que o programador pode utilizá-las sem perder o controlo de execução. Ver figura: 2.19, 2.20, 2.21 e tabela 2.1.

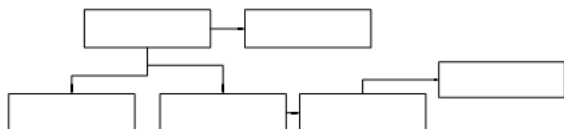


Figura 2.19: Desenvolvimento total

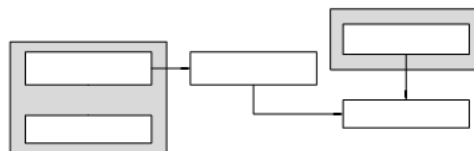


Figura 2.20: Desenvolvimento utilizando bibliotecas

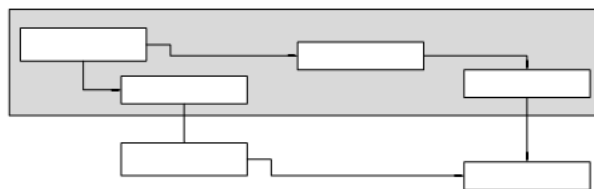


Figura 2.21: Desenvolvimento utilizando *frameworks*

Tabela 2.1: Biblioteca vs *Framework*

Biblioteca	<i>Framework</i>
Classes instanciadas pelo cliente	
Cliente chama funções	
Não possui controlo de fluxo	Controla o fluxo de execução
Não tem interacção pré-definida	Define a interacção entre os objectos
Possui comportamento <i>default</i>	Possui comportamento <i>default</i>

Frameworks JavaScript

As *frameworks* do tipo cliente disponibilizam um vasto conjunto de funções que permitem manipular dinamicamente o conteúdo/estilo da página *web*. As alterações da página *web* são efectuadas do lado do cliente funcionando em ambiente idêntico a um *desktop* e, assim, diminuem a quantidade de pedidos efectuados ao servidor.

Como referido anteriormente, podem identificar-se duas fases distintas na utilização da ferramenta: (1) o desenvolvimento do modelo e (2) a utilização do mesmo. Em ambas as situações é fundamental a disponibilidade de um conjunto de funcionalidades para tornar a utilização da ferramenta simples para o utilizador. É neste contexto que a utilização de *frameworks* consiste numa vantagem no desenvolvimento da ferramenta, uma vez que estas disponibilizam um vasto conjunto de funcionalidades que podem facilitar o papel do programador.

No sentido de clarificar as situações em que as *frameworks* podem ser utilizadas, são apresentadas um conjunto de funcionalidades utilizadas na ferramenta de criação de modelos. Estas são apresentadas pelas diferentes fases referidas.

Fase 1 - Desenvolvimento do Modelo

- *Tabbed Pane* - menus;
- *Drag Drop* - posicionamento gráfico;
- *Mouse Resizable* - dimensionamento gráfico;

- *Container* - divisões com propriedades de configuração;
- *Color Picker* - manipulação da cor;
- AJAX.

Fase 2 - Utilização do Modelo

- *Date Picker* - calendário.

As funcionalidades referidas são comuns a várias *frameworks* JavaScript como se apresenta ao longo desta secção.

Importância e Utilidade das *Frameworks*

As principais razões que motivaram a utilização de uma *framework* no desenvolvimento desta ferramenta foram:

- Reutilização de componentes previamente desenvolvidos, podendo-se assim reduzir o tempo e o esforço para o desenvolvimento do *software*;
- Redução do código desenvolvido e testado, sendo que com esta solução é necessário desenvolver apenas os pontos de interligação (ver figura: 2.21);
- As *frameworks* são periodicamente actualizadas com novas funcionalidades e correcções;
- As *frameworks* JavaScript apresentam suporte para vários clientes *web*.

Tipos de *Frameworks* JavaScript

Segundo Milfont [30] existem três tipos principais de *frameworks* JavaScript:

1. *Javascript Multipurpose*: fornecem componentes sobre o conjunto de todas as tecnologias *web* no lado cliente, e mecanismos de acesso ao lado servidor, como o encapsulamento do tratamento dos dados;

2. *Javascript Remote*: especialistas no encapsulamento do mecanismo de troca de objectos entre as camadas físicas;
3. *Javascript Specialized*: especialistas somente num determinado comportamento ou mecanismo do conjunto de tecnologias *web*, como por exemplo: especialistas em efeitos.

A *Framework - Javascript Multipurpose* é o tipo de *framework* utilizada na ferramenta desenvolvida neste projecto, uma vez que este tipo de *framework* disponibiliza funcionalidades em todas as tecnologias *web* do lado do cliente.

Este tipo de *frameworks* estão organizadas por forma a que sejam incluídas apenas as funcionalidades estritamente necessárias. Assim, é possível fazer a extensão a novas funcionalidades que actualmente não se utilizam, de forma simples e rápida. Por exemplo: utilizando uma *framework* específica para produzir efeitos/animações, esta não poderia ser utilizada posteriormente para realizar comunicações assíncronas (AJAX). Desta forma, seria necessário utilizar uma nova *framework* que disponibilize ambas as funcionalidades ou então uma nova *framework* em conjunto.

Frameworks Analisadas

Segundo Reisig [38] existe um vasto conjunto de *frameworks*, sendo que na 2.2 apresenta-se as mais utilizadas.

Tabela 2.2: *Frameworks*

Open Source	Comercial
Prototype	BackBase
Yahoo UI	SmartClient
Dojo	
jQuery	

A escolha recai sobre uma *framework* de cariz não comercial, sendo um requisito no desenvolvimento da ferramenta. Desta forma, a análise visa apenas *frameworks open*

source (YUI, Dojo, Prototype e jQuery).

Yahoo UI

A Yahoo! *User Interface (YUI) Library* é um conjunto de utilitários e controlos, escrito em JavaScript, para construir aplicações *web* interactivas usando técnicas como *scripting* DOM, DHTML e AJAX. A biblioteca YUI também inclui vários recursos de CSS. Todos os componentes da biblioteca YUI encontram-se disponíveis como *open source*, e para qualquer tipo de utilização sob uma licença BSD [46].

Dojo

Dojo é um *open source* DHTML *toolkit* desenvolvido em JavaScript. O Dojo visa ultrapassar alguns problemas do DHTML de longa data que impediram o desenvolvimento de aplicações *web* dinâmicas. Disponibiliza controlos por eventos, I/O API's e uma linguagem genérica, formando a base de um poderoso ambiente de programação. Dojo construiu um processo de ajuda para otimizar o JavaScript, possibilitando agrupar um conjunto de arquivos e reutiliza-los através de "perfis" [12].

Prototype

Prototype é uma *framework* JavaScript que facilita o desenvolvimento de aplicações *web* dinâmicas. Define-se como sendo de fácil utilização e, cada vez mais, constitui uma escolha para as equipas de desenvolvimento de aplicações *web* [37].

jQuery

jQuery é uma biblioteca JavaScript que permite simplificar a forma de manipular os documentos HTML, controlar eventos, executar animações e disponibilizar AJAX para as páginas *web*. jQuery é projectado para alterar a forma de programar em JavaScript [26].

Características das *Frameworks*

Para o desenvolvimento da ferramenta é necessário que a *framework* escolhida disponibilize um conjunto de características específicas. As *frameworks* do tipo *Javascript Multipurpose* disponibilizam uma enorme quantidade de funcionalidades, contudo as características necessárias são as seguintes:

- DOM;
- Eventos;
- Animações;
- AJAX.

DOM

DOM é uma plataforma, uma linguagem que permite programas, *scripts* aceder e actualizar dinamicamente o conteúdo, estrutura e o estilo do documento [44].

As *frameworks* YUI, Dojo, Prototype e jQuery apresentam suporte para DOM como é exemplificado. Apresenta-se de seguida dois exemplos de cada *framework*: o primeiro exemplifica como alterar uma propriedade de um elemento e o segundo permite aceder a uma propriedade.

YUI

- 1 YAHOO.util.Dom.setStyle('elementoid', 'opacity', 0.5);
- 2 var valor = YAHOO.util.Dom.getStyle('elementoid', 'opacity');

Dojo

- 1 dojo.query('elementoid').style('opacity',0.5);
- 2 var valor = dojo.query('elementoid').style.opacity;

Prototype

- 1 Element.addClassName('elementoid', 'pending');

```
2 var valor = Element.getHeight('elementoid');
```

jQuery

```
1 $('elementoid').css('color','red');
```

```
2 $('elementoid').color;
```

Eventos

Os eventos são as formas de controlar as acções do utilizador. Quando se interage com uma página *web* produzem-se vários eventos, podendo estar associado a estes um conjunto de operações.

De seguida descreve-se como pode ser definido um evento em cada uma das *frameworks*.

YUI

```
YAHOO.util.Event.addListener('elemento', 'click', function() {alert('click');})
```

Dojo

```
dojo.event.connect('elemento', 'click', function() {alert("click");})
```

Prototype

```
Event.observe('elemento', 'click', function() {alert('click');})
```

jQuery

```
$('elemento').click(function() { alert('click'); })
```

AJAX

AJAX não é uma nova linguagem de programação, mas uma técnica para criar melhores, mais rápidas e mais interactivas as aplicações *web*. Com AJAX, o JavaScript pode comunicar directamente com o servidor, utilizando o JavaScript XMLHttpRequest *object*. Desta forma, o JavaScript pode trocar informação com um servidor *web*, sem a necessidade de fazer *re-load* à página. AJAX utiliza transferência assíncrona de dados entre o

cliente e o servidor, permitindo que as páginas *web* solicitem pequenas informações ao mesmo, em vez da página completa [44].

De seguida apresenta-se como é possível utilizar AJAX utilizando as várias *frameworks*.

YUI

```
var request = YAHOO.util.Connect.asyncRequest('get', url,  
function(resposta) {  
    YAHOO.util.Dom.id('elementoid').innerHTML = resposta  
});
```

Dojo

```
dojo.io.bind({  
    url: 'url',  
    method: 'get'  
    load: function(tipo, dados) {  
        dojo.byId('elemento').innerHTML = dados;  
    }  
})
```

Prototype

```
new Ajax.Request('url',{  
    method:'get',  
    onSuccess: function(dados){  
        var resposta = dados.responseText or 'Falhou...';  
        alert(resposta);  
    },  
    onFailure: function(){ alert('Falhou...') }
```

```
});  
  
jQuery  
  
$.ajax({  
    url: 'url',  
    type: 'get',  
    error: function(){ alert('Falhou...'); },  
    success: function(dados){ alert(dados); }  
});
```

Animações

A *framework* deverá apresentar um conjunto de efeitos/animações por forma a poderem ser utilizadas na ferramenta de desenvolvimento de modelos:

- *Drag Drop*: funcionalidade que permite arrastar elementos através da página;
- *Resize*: funcionalidade que permite dimensionar elementos graficamente;
- *Color Picket*: é uma funcionalidade que permite escolher uma cor, através de diferentes formas (ver figura 2.23);
- *Date Picket*: é uma funcionalidade que apresenta um calendário sobre o qual se escolhe uma data (ver figura 2.23);
- *Tabbed Pane*: é um conjunto de menus que são acedidos através *tabs* (ver figura 2.24);

Efectuou-se uma análise comparativa relativamente às animações necessárias disponíveis em cada *framework*. A tabela 2.3 apresenta essa mesma análise.

Através da tabela comparativa, pode verificar-se que na Prototype não estão disponíveis todas as animações desejadas. Pode concluir-se assim que, a Prototype não apresenta

Tabela 2.3: Efeitos/Animações das *frameworks*

	Prototype	Dojo	YUI	jQuery
<i>Drag Drop</i>	x	x	x	x
<i>Resize</i>		x	x	x
<i>Color Picket</i>		x	x	x
<i>Date Picket</i>		x	x	x
<i>Tabbed Pane</i>		x	x	x

os requisitos necessários para ser utilizada no desenvolvimento da ferramenta, pelo que está assim excluída a hipótese da sua utilização.

Cliente *web*

Além das funcionalidades apresentadas, a *framework* deverá apresentar suporte à grande maioria dos clientes (*browsers*). As *frameworks* funcionam satisfatoriamente nos principais navegadores *web* (Firefox e IE), contudo apresentam falhas no seu funcionamento quando são empregados navegadores de utilização menos frequentes como o caso do Konqueror.

A *framework* YUI apresenta-se bastante modular e robusta (está em utilização em yahoo.com inspirando confiança), apresentando um funcionamento bastante satisfatório, mesmo quando utilizada em navegadores menos comuns. Esta *framework* funciona correctamente nos clientes *web* Inetnet Explore 6⁺, Firefox 2⁺, Opera 9⁺ e Safari 2⁺.

A *framework* Dojo tem vindo a diminuir o suporte para clientes Safari.

A Prototype não apresenta suporte para clientes Opera.

A *framework* YUI apresenta suporte para um maior número de clientes *web* que as restantes analisadas.

Documentação

Todas as *frameworks* apresentadas disponibilizam alguma documentação. A YUI apresenta uma enorme vantagem relativamente às restantes *frameworks*, uma vez que esta disponibiliza um vasto conjunto de exemplos bem documentados. Documentação relativa à explicação funcional dos vários exemplos assim como a documentação do código.

Ao longo desta sub-secção apresentou-se um estudo sobre algumas *frameworks* com a finalidade de verificar qual a melhor escolha para o desenvolvimento da ferramenta. Verificou-se que ambas as *frameworks* apresentam características muito idênticas excepto a Prototype que não disponibiliza funcionalidades gráficas (efeitos/animações).

Relativamente aos clientes *web* sobre os quais as *frameworks* podem ser utilizadas verifica-se que a YUI e jQuery apresentam uma maior gama de clientes que suportam as suas funcionalidades.

Além destas considerações verifica-se que a YUI apresenta uma melhor documentação em comparação com as restantes *frameworks* analisadas.

Assim, depois das considerações apresentadas optou-se pela *framework* que melhor se adapta às necessidades, a YUI.

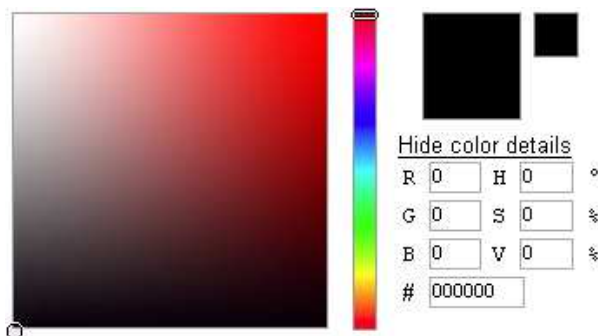
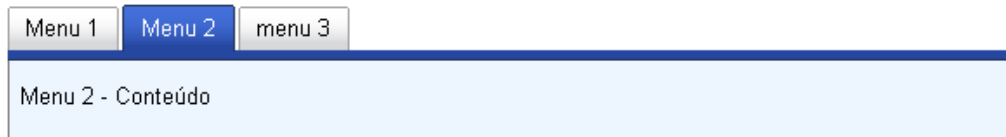


Figura 2.22: Exemplo *Color Picket*



Figura 2.23: Exemplo *Date Picket*

Figura 2.24: Exemplo *Tabbed Pane*

Exemplo - Calendário

De todas as características necessárias para desenvolvimento da ferramenta de criação de modelos apresenta-se um exemplo básico relativo à utilização de um calendário (*Date Picket*) através da *framework* YUI.

Para utilizar o calendário é necessário uma DIV onde este é colocado. A DIV deve ter um ID único que coincide com o ID passado ao construtor do calendário.

```
<div id="container"></div>
```

O calendário pode ser instanciado através do bloco de código seguinte:

```
<script>
    exampleCalendar.init = function()
        exampleCalendar = new YAHOO.widget.Calendar("container");
        exampleCalendar.render();
        YAHOO.util.Event.onDOMReady(exampleCalendar.init);
</script>
```

Neste exemplo o construtor recebe um parâmetro, o id da divisão onde fica disponível o calendário.

O calendário resultante está representado na figura 2.23

2.5 Reporting Tools

Uma das funcionalidades apresentadas na ferramenta, é a capacidade de produzir relatórios com base em informação complexa armazenada em base de dados. Para um melhor enquadramento com este tipo de funcionalidades são apresentadas as principais características deste tipo de ferramentas.

2.5.1 Comprar ou Desenvolver

Existe uma grande variedade de relatórios estatísticos, e quer comprar quer desenvolver uma ferramenta que vá de encontro às necessidades da empresa, está fortemente dependente do tipo de exigências da mesma. Segundo keydata [1] a decisão é baseada no seguinte:

Quantidade de Relatórios

Quanto maior o tipo de relatórios que a ferramenta pode criar, mais vantagens terá. Não apenas porque as ferramentas de relatórios normalmente tornam mais fácil a criação de novos relatórios (oferecendo componentes reutilizáveis), mas também porque tem incorporado um sistemas de gestão de relatório para tornar mais fácil a manutenção e funções suporte.

Relatório num Único Formato

Se os relatórios apenas serão enviados numa única modalidade (por exemplo, apenas e-mail, ou apenas apresentados no *browser*), deverá ser considerada a possibilidade de desenvolvimento da ferramenta a partir do zero. No entanto, se os utilizadores terão acesso a relatórios através de uma variedade de canais, fará sentido investir numa ferramenta onde já vem incluídos a distribuição desses modos.

Criação de Relatórios *Ad-hoc*

Poderão os utilizadores criar os seus próprios relatórios *ad-hoc*? Se assim for, a aquisição deste tipo de ferramenta poderá ser uma vantagem. Os distribuidores deste tipo de ferramenta têm à priori um conhecimento das características que são mais importantes que os utilizadores necessitam de criar relatórios *ad-hoc*. A segunda razão é que para a criação de relatórios *ad-hoc* é necessária uma grande quantidade de dados, o que é uma tarefa difícil quando se está a começar do zero.

Os dados são inúteis se apenas permanecem numa base de dados. Para os dados serem verdadeiramente importantes é necessário que estes sejam apresentados ao utilizador de uma forma que possa ser interpretada. Como resultado, a apresentação é de elevada importância.

A maioria dos fornecedores OLAP (*Online Analytical Processing*) já têm um *front-end* apresentação que permite aos utilizadores usarem relatórios pré-definidos ou criar relatórios *ad-hoc*.

OLAP é uma abordagem tecnológica para gerar respostas rápidas a consultas analíticas de natureza tipicamente dimensional. A tecnologia OLAP é parte de uma categoria mais abrangente, *business intelligence*, que também inclui *data warehouse* (que por sua vez inclui ETC (Extracção Transformação e Carga) e *data mining*. Aplicações típicas de OLAP são relatórios de negócios, *marketing*, relatórios de gestão, BPM (*Business Process Management*), orçamento e previsão, relatórios financeiros e áreas similares. O termo OLAP foi criado como uma ligeira variação de um termo tradicional em base de dados, OLTP (*On Line Transaction Processing*) [2].

2.5.2 Funcionalidades

Existem também vários tipos de ferramentas para produzir relatórios. De qualquer forma, deve-se prestar atenção aos seguintes pontos na avaliação de ferramentas de relatórios [1]:

Capacidade de Ligação à Fonte de Dados

Em geral, existem dois tipos de fontes de dados. Uma é a relação de dados outra é a fonte de dados OLAP multidimensional. Actualmente, ambas constituem bons investimentos. Muitos vendedores deste tipo de ferramentas dizem que oferecem ambas as opções, mas decorrente de uma observação mais minuciosa, é possível que a ferramenta seja especialmente boa para um tipo, mas proceder à ligação a outro tipo de fonte de dados, torna-se um difícil exercício de programação.

Programação e Distribuição Capacidades

Tendo por base um cenário plausível *data warehousing* utilizado por altos executivos, todos eles na manhã de segunda-feira, conferem os mais importantes números da semana anterior (dizem os números de vendas), e para a partir deles suprimirem as suas necessidades. Toda a concepção *ad-hoc* e o exercício em torno das capacidades, não lhes interessará, uma vez que eles nem sequer dão importância a estes aspectos.

Com base no cenário anteriormente descrito, a ferramenta de relatórios deve assim incluir a programação e distribuição de tarefas. Os relatórios semanais são programados para serem publicados na segunda-feira de manhã, e os relatórios resultantes são distribuídos para os executivos experientes ou pelo e-mail ou pela publicação na *web*, sendo estes meios os mais vantajosos.

Características de Segurança

Uma vez que as ferramentas de relatórios, semelhante às ferramentas OLAP, são orientadas para um elevado número de utilizadores, garantir que as pessoas vêem apenas aquilo que é suposto é muito importante. A segurança pode residir ao nível do relatório, ao nível de pasta, ao nível da coluna, ou mesmo ao nível de células individuais. De um modo geral, todas as ferramentas de relatórios têm já predefinidas essas capacidades. Além disso, eles têm uma especificação de segurança semelhantes aos comuns protocolos de *login*. No entanto, existem casos em que as grandes corporações têm desenvolvido os

seus próprios mecanismos de autenticação do utilizador. Para estes casos, uma integração perfeita entre a ferramenta e a *in-house* autenticação interna da cooperação pode exigir algum trabalho.

Personalização

Todos nós já nos sentimos frustrados quando gastamos uma enorme quantidade de tempo com algumas ferramentas do Office apenas para que o relatório tenha uma boa apresentação. É sem dúvida uma perda de tempo, mas infelizmente, é um mal necessário. De facto, muitas vezes os analistas desejariam desenvolver um relatório já predefinido pela ferramenta de relatórios adaptando-o às suas apresentações ou relatórios que irão ser entregues aos seus superiores. Se a ferramenta de relatórios oferecer uma forma fácil de pré-definir os relatórios para que estes assumam o padrão corporativo, o trabalho dos analistas será facilitado bem como o tempo necessário para o desenvolvimento do relatório será rentabilizado.

2.5.3 Constituintes do Relatório

A produção de relatórios é composta por quatro partes principais:

- Dados;
- Transformação de dados - relatórios de dados, resumidamente, filtrada e agrupados para se ajustar as necessidades do utilizador;
- Lógica de Negócio - lógica para converter dados brutos em informações úteis para o utilizador;
- Apresentação.

2.5.4 Ferramentas

Algumas das ferramentas de criação de relatórios disponíveis no mercado são:

- Crystal Reports;
- Cognos;
- Oracle Business Intelligence Discoverer;

Capítulo 3

Plataformas Tecnológicas

3.1 iPortlaDoc

O iPortalDoc é um sistema de gestão de documentação e *workflow* para empresas e instituições, que permite aos seus utilizadores a gestão dos fluxos dos documentos como também, simplesmente, proceder ao seu arquivo para posterior gestão. Para um bom funcionamento do sistema, é vital que este seja bem organizado. Este sistema de gestão de documentação encara a informação digital como um substituto do papel, e não uma tecnologia adjacente [24].

As características mais importantes do iPortalDoc são:

- Facilidade de utilização, formação reduzida;
- Integrada no ambiente de trabalho dos utilizadores;
- Redução do *paperware* e controlo de processos com utilização de *workflows*;
- Segurança, no acesso e acções sobre documentos.

3.1.1 Arquitectura

O iPortalDoc não funciona independentemente, ou seja, o iPortalDoc funciona em colaboração com outro sistema, a IP Brick, utilizando o primeiro um conjunto de serviços disponibilizados neste sistema operativo (IPBrick). Como pode ser observado na figura 3.1 o iPortalDoc necessita de um sistema de gestão de base de dados de suporte ao sistema de ficheiros, para armazenar os documentos. Além disto, está integrado com um servidor *web*, servidor de correio electrónico, servidor de ficheiros e um servidor de informação e gestão de domínios.

É na IPBrick que se faz a inserção e gestão dos utilizadores do iPortalDoc, bem como dos grupos aos quais eles pertencem. Também é aí que serão criados os registos das entidades (através da aplicação IP Contactos), e respectivos contactos com as quais a instituição interage, e que também são necessários durante a actividade no iPortalDoc.

O iPortalDoc trabalha integrado, para além dos componentes básicos acima referidos, com um servidor *web* (HTTP), um servidor de correio electrónico próprio (SMTP/-POP/IMAP), um servidor de ficheiros (SMB) e um servidor de informação e gestão de domínios (LDAP). Está ainda coordenado com os serviços de gestão de nomes lógicos (DNS) e atribuição de endereços IP às estações de trabalho (DHCP), entre outros.

Este conjunto de serviços de Intranet que acompanham o iPortalDoc constituem a IPBrick, que pode ser usada na Intranet da instituição ou integrar-se com esses serviços caso já existam.

3.1.2 Segurança

Ao nível de rede a segurança no iPortalDoc, como serviço de rede de comunicações que é, está directamente ligada à segurança existente para os utilizadores na Intranet, estejam eles definidos na IPBrick, ou a esta recorra a servidores de *Active Directory* presentes na rede.

Ao nível de gestão documental, a segurança está associada aos perfis de utilizadores

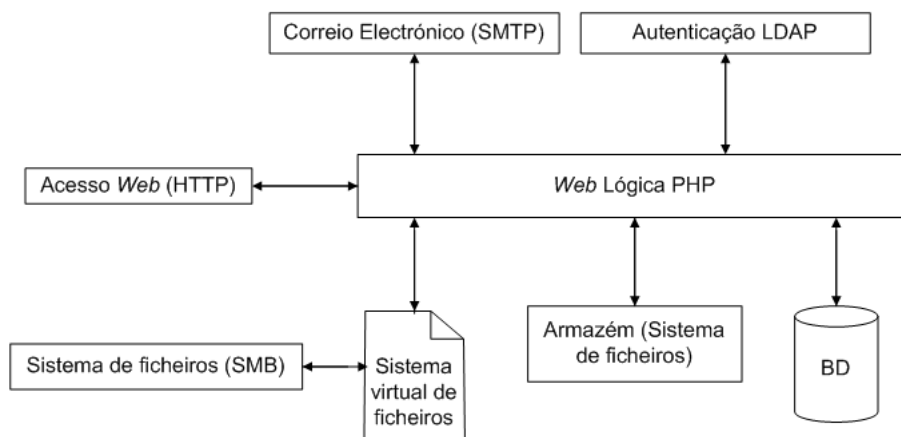


Figura 3.1: Arquitectura do iPortalDoc

definidos pelo administrador, aos estados do fluxo e utilizadores associados ao fluxo. Por exemplo, enquanto o fluxo não termina, só os utilizadores que participam no fluxo é que tem acesso ao documento, de acordo com o perfil e acções que podem executar no documento. Esta visão da segurança avançada tem portanto três níveis:

- Segurança de rede (autenticação de acessos através de qualquer interface de comunicações)
- Perfil de utilização no gestor (gestão dos acessos aos documentos);
- Presença no fluxo e acções disponíveis (controlo das acções a executar sobre os documentos).

Os perfis podem ser definidos e adaptados de acordo com os requisitos de cada instituição, através da combinação de um conjunto de permissões: criação, leitura, eliminação para documentos, secções e documentos na secção. De realçar a possibilidade de os documentos poderem herdar as permissões gerais dos documentos na secção, isto é essencial para aumentar o desempenho do sistema em secções com milhares de documentos.

O iPortalDoc e a IPBrick estão integrados com sistemas biométricos, o que permite que autenticação para aceder ao gestor documental precisam da impressão digital, assim

como foram definidas acções de aprovação de documentos que implicam autenticação biométrica.

3.1.3 Permissões

Os documentos no SGDW são armazenados por uma estrutura hierárquica. A hierarquia tem uma directoria raiz e, dentro desta, podem ser criadas outras directorias e assim sucessivamente. Esta organização de directorias é configurada de acordo com a necessidade da entidade utilizadora e, desta forma, pode manter todos os documentos organizados. A árvore de directorias é sempre apresentada ao utilizador, afim deste poder navegar facilmente através da mesma. Esta será apresentada de acordo com utilizador em questão, ou seja, a hierarquia é apresentada de acordo com as permissões de cada utilizador.



Figura 3.2: Hierarquia dos documentos

Após o utilizador efectuar a autenticação no sistema, são-lhe facultadas diferentes funcionalidades, dependentes das permissões associadas ao perfil que este tem atribuído em cada directoria.

Cada utilizador pode ter perfis distintos, dependendo da directoria, ou seja, um utilizador numa directoria pode ser Coordenador e noutra apenas Leitor. O facto de serem atribuídos perfis nas directorias existentes, permite gerir o acesso aos documentos existentes nessas mesmas directorias. Existem um conjunto de perfis pré-definidos:

- Administrador: um utilizador ao qual está atribuído um perfil com permissões totais sobre todas as funcionalidades do sistema. O administrador pode, entre outros, configurar e alterar os *workflows*, criar, remover, ou alterar grupos de utilizadores; introduzir, alterar ou remover dados de um utilizador do sistema; criar, remover ou alterar perfis;
- Coordenador: um utilizador que disponha de um perfil que lhe permita efectuar todas as funcionalidades numa determinada directoria (criar, alterar, remover documentos/directorias), é referido como sendo o Coordenador desta. O coordenador pode associar perfis e *workflows* a utilizadores, na directoria que coordena. Um utilizador que apresente este perfil, pode visualizar todos os documentos, independentemente do seu estado no *workflow*;
- Sub-Coordenador: um utilizador que dispõe deste perfil tem permissão para efectuar todas as funcionalidades com excepção de apagar directorias. Um Sub-Coordenador pode associar perfis e *workflows*, na directoria que coordena;
- Leitor Absoluto: Um utilizador que possua este perfil tem permissão para ler todos os documentos das directorias existentes no sistema, mesmo que não tenha intervenção nos mesmos. No entanto, não lhe é permitido criar, alterar ou remover documentos ou directorias;
- Editor: para além das permissões associadas a um Leitor, este ainda pode inserir documentos;
- Navegador: tem apenas permissão para ver as directorias, não podendo criar nem apagar. Não tem qualquer acção sobre os documentos.

Na figura 3.3 está representado o menu para gestão de perfis de utilizadores, onde está representado o perfil Administrador.

Em cada directoria são associados utilizadores com o tipo de perfil pretendido. É possível associar um grupo de utilizadores à directoria ou então fazer a associação individual de cada utilizador.

Gestão de Perfis

Perfil:	Super User	
Nome:	Super User	
		Super User ☒
		Coordenador ☐
		Leitor absoluto ☒
Permissões:	Documentos	Secções
	Ler ☒ Criar ☒ Apagar ☒	Ler ☒ Criar ☒ Apagar ☒

Figura 3.3: Perfis

As directorias possuem utilizadores associados, sendo que em cada directoria, os utilizadores podem ter um conjunto de tipos de documentos associados. A associação do tipo de documento ao utilizador nessa directoria pode ser aplicado pelas restantes sub-directorias, ou seja, os tipos de documentos associados à directoria raiz são aplicados recursivamente por todas as suas directorias.

Cada utilizador pode ter associado vários *workflows*, ou nenhum, dependendo do seu perfil nessa directoria. Por exemplo: se for associado um utilizador com o perfil Leitor Absoluto, não será possível associar *workflows* ao utilizador para essa directoria, uma vez que nesta situação o utilizador poderá ler documentos mas não pode efectuar operações sobre os mesmos (criar, alterar). Como acontece na associação de tipos de documentos, quando é efectuada a associação, pode ser efectuada a um grupo de utilizadores ou então individualmente. A associação também pode ser recursiva pelas sub-directorias.

3.1.4 Modelos de Documentos

A ferramenta de desenvolvimento de modelos de documentos funciona de forma integrada com SGDW. Ao aceder a esta ferramenta, é apresentada uma listagem dos modelos já desenvolvidos no caso da existência destes. Para criar um novo modelo é apresentado um formulário onde deve ser especificada a seguinte informação (ver figura 3.5):

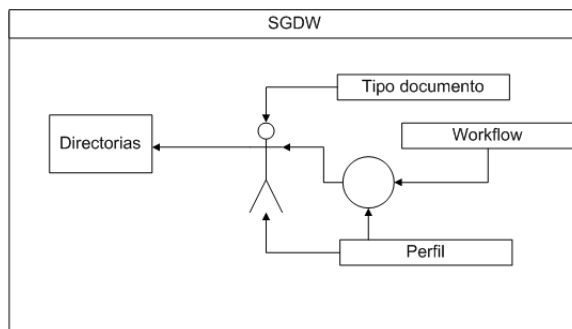


Figura 3.4: Permissões nas directorias

- Descrição do modelo;
- Estatísticas: esta funcionalidade permite que a informação introduzida no formulário de criação do documento PDF seja armazenada na base de dados para posterior utilização.
- Importação: os modelos dos documentos podem ser importados, ou seja, estes modelos podem ser produzidos num SGDW e posteriormente utilizados noutro SGDW.

Gerador de templates de documentos [Docs.](#)

[Listagem](#) | [Importar](#) | [Template](#)

Template:

Para estatísticas: ☒
 (Se seleccionada, os dados de cada documento gerado serão guardados na BD.)

Figura 3.5: Introdução de modelos de documentos

Para modificar um modelo anteriormente configurado, o utilizador deve seleccionar o novo modelo pretendido. Nesta situação, será verificado se existe algum documento deste tipo, ou seja, verifica a existência de algum documento que tenha sido introduzido através do formulário do modelo que se pretende editar, e, em caso afirmativo, o utilizador é notificado.

Os documentos introduzidos no SGDW têm de passar por um processo de classificação. Nesse processo, o utilizador deverá preencher o formulário devidamente para facilitar posteriores operações sobre o documento e, desta forma, minimizar os processos de pesquisa, minimizando consequentemente também o tempo de resposta, quando é utilizado o motor de busca.

Neste processo de classificação, uma das opções do formulário é o tipo de documento. Mediante a selecção deste, será apresentado o formulário do documento caso exista um modelo associado ao tipo de documento seleccionado. No caso de não existir nenhuma associação entre o tipo de documento e o modelo de documento, a introdução do documento é efectuada através da opção de *upload*.

3.1.5 Tipos de Documentos

O tipo de documento é uma informação que aparece sempre que o utilizador introduzir um documento no SGDW, sendo esta uma característica importante de classificação para os documentos introduzidos no SGDW. O utilizador ao aceder a esta funcionalidade pode criar, alterar ou remover tipos de documentos desde que no formulário de criação de tipos de documentos sejam especificados um conjunto de campos:

- Descrição do documento;
- Sigla associada ao tipo de documento;
- Código: esta opção permite definir como irá ser efectuada a gestão dos códigos do tipo de documento, sendo que o utilizador pode configurar o código como entender de acordo com um conjunto opções disponíveis;
- Template de geração automática: nesta opção, permite ao utilizador indicar qual o formulário que está associado a este tipo de documento, ou então indicando que o documento é introduzido através de *upload* para o sistema;
- Permissão - Tipo Doc.: esta opção, quando seleccionada, permite alterar o tipo de documento e código atribuído a um documento já inserido;

- Permissão - Info: permite definir a possibilidade de alterar a informação de classificação do documento no decorrer do *workflow*.
- Alterar informação: permite definir a possibilidade da informação do documento ser alterada;
- Tipo Registo: esta opção define que o documento nunca poderá ser actualizado.

Na figura 3.6 está representado o formulário de criação de tipos de documento.

Gestão dos Tipos de Documentos

Tipos de Documentos:

Descrição:

Sigla:
(Sigla que irá ser utilizada na geração do código do documento)

Código:
(Formato do código dos documentos)

[Gestão Códigos](#)

Template de geração automática:

Permissão - Tipo Doc.: ☐
(Ao estar seleccionada, permite alterar o tipo de documento e código atribuído a um documento)

Permissão - Info: ☒
(Ao estar seleccionada, permite a alteração da informação do documento, durante o workflow.)

Alterar informação: ☐
(Ao estar seleccionada, impede que a informação do documento seja alterada)

Tipo Registo: ☐
(Ao estar seleccionada, impede que o documento seja actualizado)

Figura 3.6: Tipos de documento

3.1.6 Introdução de Documentos

Para introduzir qualquer documento o utilizador deve verificar a directoria na qual deve ser introduzido. Ao introduzir um documento o utilizador acede a uma interface que possibilita fazer a classificação do mesmo para posterior introdução.

O processo de introdução de documentos baseados em qualquer modelo gerado pelo utilizador consiste num conjunto de três operações principais:

1. Classificação;
2. Preenchimento do formulário do modelo do documento;
3. Submissão.

Classificação

Qualquer documento a ser introduzido no SGDW tem de passar por um processo de classificação. Cada documento deverá, assim, ser devidamente classificado para que possa ser encontrado facilmente através do motor de busca (pesquisa). O formulário de classificação de cada documento é apresentado na figura 3.8. Para efectuar este processo existem um conjunto de campos que são obrigatórios e um outro conjunto de carácter opcional.

Os seguintes campos são obrigatórios:

- **Tipo de documento:** ao ser atribuído um tipo de documento é gerado automaticamente um código para o documento em questão;
- **Workflow:** o fluxo de trabalho que o documento a inserir terá de percorrer (ver secção 2.2). Na figura 3.7 está representado um exemplo de um *workflow* do iPortalDoc;
- **Título:** corresponde ao título com o qual o documento será listado;
- **Ficheiro** é o campo onde o utilizador faz o *upload* do documento para o sistema. No caso do tipo documento a ser introduzido estiver associado a um modelo de documento, esta opção não está disponível;

Além dos campos obrigatórios, existem outros campos que ajudam a complementar a descrição de um documento. São eles:

- **Tipo de Entidade,** onde o utilizador selecciona o ramo da entidade ao qual está associado o documento. Após a escolha do tipo de entidade, surgirá o campo **Entidade** onde se especifica qual a entidade dentro do tipo seleccionado anteriormente;

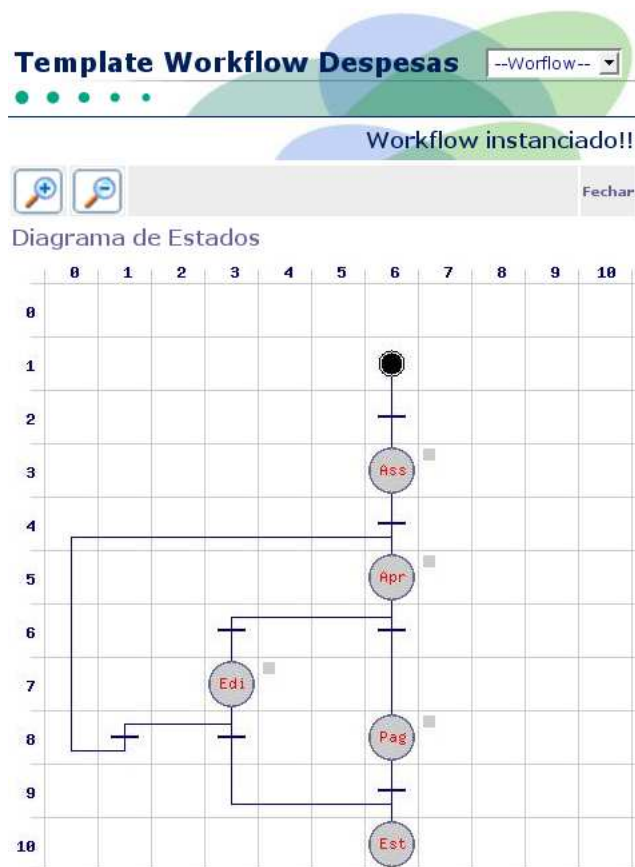


Figura 3.7: Exemplo de um *workflow* no iPortalDoc

- **Assunto:** nesta opção é possível inserir o assunto a que se refere o documento que vai ser introduzido;
- **Ordem:** através desta funcionalidade, o utilizador pode especificar a ordem em que quer que apareça o documento dentro da secção em que ele será introduzido.
- **Edição:** Neste campo o utilizador pode definir o número de edição do documento;
- **Valor:** Através desta opção o utilizador pode definir um valor para o documento introduzido, como por exemplo, o valor monetário de uma proposta;
- **Elaborado em:** é o campo onde se coloca a data de criação do documento;
- **Sumário e Descrição** são campos onde o utilizador pode expor o conteúdo do

documento, sendo que, por norma, o sumário é mais sucinto do que a descrição;

- **Localização física:** é o campo onde o utilizador deve referir o local onde se encontra o formato palpável do documento (por exemplo, a localização no arquivo de um documento em papel);
- **Palavras-chave:** é o campo destinado à colocação de palavras que melhor descrevem o documento e que poderão ser úteis para encontrar o documento em pesquisas por palavra-chave;
- **Centro de Custo e Sub-Centro de Custo:** como é normal, as empresas estão divididas em diferentes secções, cada uma com os seus gastos específicos. Assim, os documentos relacionados com os custos de uma área delimitada da organização (por exemplo facturas) podem ser classificados através desta opção;
- **Network:** esta funcionalidade surge como mais um campo de classificação do documento, possibilitando a especificação do mesmo, tendo em conta o projecto ao qual está associado numa empresa.

A informação introduzida no formulário de classificação pode ser utilizada no formulário do modelo do documento, uma vez que o utilizador tem ao seu dispor opções para utilização desta informação.

Entidades: um dos elementos de classificação apresentado foi a Entidade. Para gerir as entidades, o sistema operativo IPBrick disponibiliza uma aplicação designada IP Contatos que permite gerir todas as entidades de uma organização, e nesta são definidos os perfis associados às respectivas entidades (sincronizados via MS Outlook ou *browser web*). Para introduzir entidades, é necessário inserir, numa primeira fase, tipos de entidades. Os tipos de entidades são categorias nas quais o utilizador pode organizar as entidades relacionadas com a sua organização (p.ex. clientes, fornecedores, parceiros, entre outros). O SGDW utiliza a informação desta aplicação para classificar os documentos, nomeadamente para preencher os campos tipo de entidade e entidade.

Introduzir documento

(na Directoria: Domain/**Modelos Docs/**)

Tipo de Entidade:	
Entidade:	
	Limpar selecção
Localização:	
Contacto:	
Assunto:	
Tipo de Doc.:	
	☐ Selecção para Mail Merge
Workflow:	
Título:	
Ordem:	
Edição:	
Código:	
Valor:	
Autor:	
Elaborado em:	25 / 3 / 2008 (dia/mês/ano) 
Geração automática:	

Figura 3.8: Introdução de um documento

Os campos disponíveis para classificação de qualquer entidade são apresentados na tabela 3.1.

Tabela 3.1: Campos de classificação entidades

Nome	Nº Contribuinte	Morada	Código Postal
Telefone	Telemóvel	Fax	e-mail
Contacto msn	<i>Web page</i>	Domínio da Empresa	NIB
País	Distrito	Concelho	Freguesia
C.A.E.	Grupo Associado	D.L.A	Observações
Tipo de entidades			

Os dados apresentados relativos à classificação de documentos e os dados de classificação das entidades podem ser utilizados nos modelos de documentos. Quando esta informação é utilizada, os campos do formulário respectivos aparecem preenchidos ao utilizador.

Preenchimento do Formulário do Modelo do Documento

Após a classificação, é apresentado um formulário para introduzir a informação que permitirá criar o documento PDF. Quando o utilizador desenvolve um modelo onde são utilizados os elementos da classificação do documento, os campos obrigatórios da classificação podem variar, ou seja, estes dependem da informação configurada. Por exemplo: quando é seleccionada o nome da entidade para ser utilizada no documento PDF, o preenchimento da entidade é obrigatório. No caso de não ser utilizado um modelo de documento para introduzir documentos, este campo não seria obrigatório mantendo-se os campos tipo de documento, *workflow*, e título como obrigatórios.

Submissão

A última fase do processo de introdução de documento é a submissão da informação do formulário do modelo do documento. Depois de classificado e preenchido o formulário, o utilizador submete a informação, sendo que toda a informação passa por um

processo de validação. No caso da informação estar correcta é criado o documento PDF e introduzido no sistema de gestão documental associado ao respectivo *workflow*.

3.2 IPBrick

Cada vez mais as empresas estão dependentes da partilha da informação e das comunicações, onde a velocidade e segurança são sempre factores importantes. A administração de redes, a gestão dos seus recursos, impressoras, ficheiros, contas de e-mail, acessos à Internet, segurança contra intrusões e outras ameaças que "espreitam" qualquer sistema que não se encontre isolado do mundo, os vírus e os *worms*, podem tornar a vida de uma empresa num suplício. A IPBrick é um servidor integrado completo, desenvolvido sobre a tecnologia Linux, e possui uma interface de gestão funcional com acesso via *web*, que permite efectuar a sua configuração sem que o seu administrador possua conhecimentos Linux ou mesmo de redes e serviços. Para além da interface funcional, a IPBrick tem ainda uma interface avançada, a partir da qual um administrador de redes e sistemas tem acesso directo a todos os seus serviços, tais como DNS (*Domain Name System*), DHCP (*Dynamic Host Configuration Protocol*), LDAP (*Lightweight Directory Access Protocol*), servidor de correio electrónico, servidor de ficheiros, gestor de projectos, VPN (*Virtual Private Network*) server, *firewall* entre outros.

A IPBrick é um sistema operativo para gestão de redes com administração simples e funcional, devido à sua interface gráfica GUI (*Graphical User Interface* - Interface Gráfica do Utilizador). Desta forma, as empresas não dependem mais de peritos informáticos para manter a espinha dorsal do negócio em funcionamento com elevada disponibilidade.

A IPBrick estabelece um padrão de estabilidade e segurança na transferência de dados. Como o centro de todo o trabalho colaborativo, pode ligar-se através do MS Outlook ou através de um *browser web*. A tecnologia baseada em Linux garante confiança, velocidade e eficiência nas transferências de dados. Com a integração do antivírus Kaspersky minimiza-se o risco de perda da integridade dos dados.

Constituída por um modelo de serviços de Intranet e de comunicações, válido em qualquer tipo de empresa, tornando assim fácil a configuração de todos os serviços sem quaisquer conhecimentos de Linux e/ou redes. As configurações mais específicas podem ser realizadas por técnicos qualificados através da interface avançada.

3.2.1 IPBrick.I

A IPBrick.I disponibiliza todos os serviços necessários de um servidor de Intranet. IPBrick.I pode ser o servidor de e-mail, ficheiros, domínio, fax, impressão e de *backup*. Para além do calendário/agenda partilhados e gestor de contactos (sincronizados via MS Outlook ou *browser web*), é possível gerir todos os projectos de um ponto central.

IPBrick.I integra-se facilmente com serviço de directório. Em termos de escalabilidade oferece um conceito *master/slave* baseado em LDAP e MS *Active Directory*.

O servidor de Intranet IPBrick.I pode ser utilizado em três modos diferentes: *master*, *slave* e *AD client*. Consequentemente os utilizadores da IPBrick, grupos e dispositivos [terminais e telefones SIP (*Session Initiation Protocol*)] podem ser definidos na IPBrick ou no Windows AD (*Active Directory*). Na Intranet, a tecnologia IPBrick é capaz de funcionar num ambiente totalmente IPBrick ou integrada com sistemas Windows.

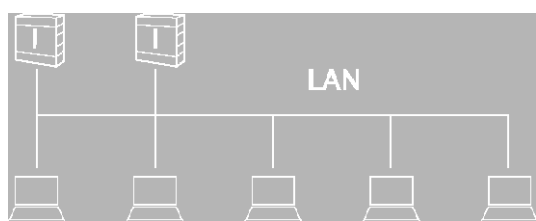


Figura 3.9: IPBrick.I inserida na LAN de uma empresa [22]

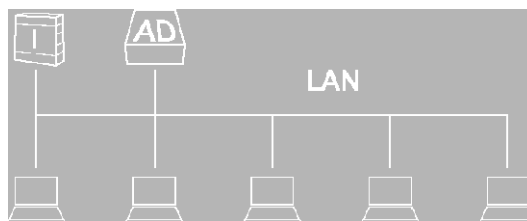


Figura 3.10: IPBrick.I na LAN de uma empresa com AD [22]

A IPBrick.I disponibiliza o seguinte conjunto de serviços, sendo estes apresentados na tabela 3.2.

Tabela 3.2: Serviços da IPBrick.I

Serviços
Servidor de áreas de trabalho individuais e de grupo
Servidor LDAP (gestão de máquinas, utilizadores e grupos)
Servidor de Domínio (com <i>roaming-profiles</i> e <i>centralized scripting</i>)
Servidor de correio electrónico
Servidor de impressoras
Servidor de base de dados
Servidor de DHCP (<i>Dynamic Host Configuration Protocol</i>)
Servidor de DNS (<i>Dynamic Network Services</i>)
Servidor de imagens de estações de trabalho
Servidor de Groupware (Calendário e Livro de Endereços)
Servidor de <i>Backup</i> (ARKEIA)
Gestor de projectos (dotProject)
Serviços Opcionais
Kaspersky Anti-Malware
Correio Electrónico (AntiVirus, AntiSpam)
Servidor de ficheiros (AntiVirus)
<i>Single sign-on</i> (Autenticação Biométrica)

3.2.2 IPBrick.C

A IPBrick.C garante a segurança na transferência de dados entre Intranet e Internet. Os dados transferidos (*download and upload*) são filtrados, analisados e geridos pela IPBrick.C. Estas operações são executadas pelos principais componentes da IPBrick.C: *Mail-Relay*, Servidor-Proxy, IDS (*Intrusion Detection System*), Antivírus e AntiSpam. A IPBrick.C controla todas as interfaces disponíveis para a Internet. O objectivo da IPBrick.C é gerir as ligações da sua empresa com a Internet. Os serviços oferecidos são: servidor *web*, servidor FTP (*File Transfer Protocol*), VPN (*Virtual Private Network*) *gateway* e telefonia VoIP (voz sobre IP).

A tecnologia VPN não só permite o fluxo de dados seguro mas também permite a es-

colha do espaço de trabalho conforme as necessidades. A IPBrick.C gere as ligações permanentes entre diversos pontos através de túneis IPSec (*IP Security*) assim como através de túneis OpenSSL (*Secure Sockets Layer* e *Transport Layer Security*) para ligações de confiança. O correio electrónico assim como o tráfego *web* é continuamente filtrado através de regras configuráveis. Relatórios com estatísticas apresentam a informação essencial em gráficos de interpretação fácil. Através do *web mail* podem ser geridos os e-mails de uma forma segura, mesmo que através de um *browser web* em qualquer lugar e momento.

A IPBrick.C pode ser colocada numa DMZ (*DeMilitarized Zone*) protegida por *firewall*, como um servidor de comunicações protegida pela *firewall*, ou mesmo como um servidor completo de comunicações e sistema de *firewall* integrado. A IPBrick.C pode importar os utilizadores, grupos e dispositivos (estações de trabalho e telefones SIP) de uma IPBrick.I ou de um sistema Windows AD. Isto significa que, instalando a IPBrick.C como servidor de comunicações, não é necessário redefinir as informações do sistema já disponíveis no servidor de Intranet da empresa.

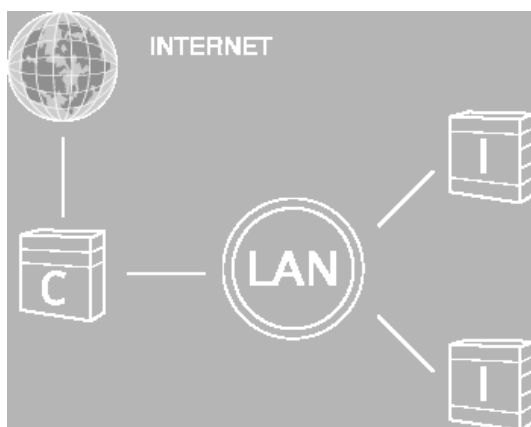


Figura 3.11: IPBrick.C numa DMZ com sistema de *firewall* [22]

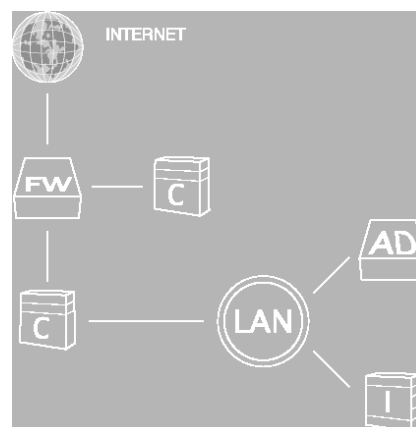


Figura 3.12: IPBrick.C numa DMZ protegida por *firewall* [22]

A IPBrick.C disponibiliza os seguintes serviços de empresa para a ligar à Internet, sendo estes representados na tabela 3.3.

Tabela 3.3: Serviços da IPBrick.C

Serviços
<i>Relay</i> de correio electrónico <i>webmail</i> Servidor <i>web</i> Servidor <i>Proxy</i> (HTTP, HTTPS, FTP com estatísticas) <i>Traffic shaping</i> <i>Port security (packet based firewalling)</i> IDS (<i>Intrusion Detection System</i>) Servidor de VoIP (RTP bases <i>routing</i>) Integração transparente com PBX (ISDN E1/BRI e linhas analógicas)
Serviços Opcionais
Kaspersky Anti-Malware Correio Electrónico (AntiVirus, AntiSpam) Servidor de ficheiros (AntiVirus) Integração transparente com PBX (ISDN E1/BRI e linhas analógicas)

3.2.3 IPBrick.GT

A IPBrick.GT complementa a solução de *software* IPBrick.IC com *hardware* seleccionado para permitir uma integração optimizada de voz e dados, esta responde às exigências da integração necessária entre a telefonia tradicional, os novos telefones VoIP SIP e fornecedores de SIP.

A IPBrick.GT implementa um *gateway* completo PSTN/VoIP, permitindo a ligação directa ao PBX (RDIS E1/Bri e linhas analógicas), a operadores PSTN, à LAN (*Local Area Network*) da empresa e à Internet. O encaminhamento inteligente permite o fluxo de voz através de VoIP, ou através do tradicional RDIS, ou mesmo linhas analógicas. A IPBrick integra uma funcionalidade de controlo de tráfego que irá dar prioridade todas as ligações de telefonia. O número de chamadas telefónicas activas e simultâneas é apenas limitado pela capacidade da ligação à Internet.

Tabela 3.4: Características da IPBrick.GT

Características
Comunicação de dados VoIP baseada em protocolos standard de Internet (SIP, RTP)
Transferência segura de dados VoIP através de túneis VPN definidos
Negociações Voice Call inteligentes para servidores VoIP estrangeiros, Fornecedores de SIP e redes de telefone tradicionais.
Controlo da ligação entre a linha RDIS/Analógica e o seu PBX
Uma evolução da migração, incorporando a herança dos telefones RDIS/Analógicos e PBX para a tecnologia VoIP
Integração com operadores de VoIP
Características avançadas de QoS com integração de gestão de largura de banda

3.2.4 IPBrick.KAV

A IPBrick.KAV é uma *appliance* de segurança baseada no *software* de base IPBrick.IC com o objectivo de proteger as empresa em quatro vertentes de segurança mais importantes da actualidade: correio electrónico, acessos *web*, segurança de rede e de Intranet.

A IPBrick.KAV é uma solução que agrega o melhor de dois mundos em termos de segurança: servidor de comunicações e *software* de segurança.

O servidor de comunicações da IPBrick.KAV impede que as estações de trabalho se liguem directamente à Internet, não permitindo que programas do tipo *trojan* estabeleçam túneis com o exterior, ou abram *backdoors* que permitem o acesso à rede da empresa a partir do exterior, eliminando os problemas causados por *trojans*. Para isso a IPBrick dispõe de um conjunto de serviços funcionando em modo *proxy*.

IPBrick.KAV foi concebida para, através de um servidor de comunicações instalada numa *appliance* desenhada com elevado nível de integração ao nível do *hardware*, do sistema operativo de rede IPBrick.IC e *software* de segurança da Kaspersky, aumentar a segurança das redes empresariais.

O *software* de segurança da Kaspersky elimina na protecção perimetral assumida pela IPBrick.KAV todos os conteúdos de *malware* (*trojans*, *worms*, *spyware*, *phishing*, vírus, entre outros) que se destinam a infectar os computadores da rede interna.

A IPBrick.KAV apresenta as seguintes características:

Tabela 3.5: Características da IPBrick.KAV

Características
Segurança <i>web</i> Anti-vírus Kaspersky, remover conteúdos perigosos existente em sites <i>Black and White lists</i> para controlo dos acessos à Internet impedindo ou limitando o acesso a certos Segurança de Mail Anti-vírus e Anti-spam da Kaspersky, limpar conteúdos perigosos enviados por correio electrónico <i>Denial of service</i>, reduz os ataques que pretendem bloquear o correio electrónico da empresa (<i>mail bombing</i>) Segurança de Rede <i>Firewall</i>, para impedir acessos não desejados à rede interna a partir do exterior VPN, aumentar nível de segurança do acesso dos utilizadores aos recursos internos Deteção de Intrusão, identificar ou alertar os administradores de acessos suspeitos Segurança de Intranet Deteção de máquinas infectadas na Intranet

A IPBrick.KAV é um elemento essencial para a segurança da empresa, mas também oferece funcionalidades para melhorar a operação da sua empresa, na interacção entre a rede interna e a Internet:

- *webmail*: para poder consultar o seu correio electrónico a partir de qualquer lugar de forma segura;
- VPN SSL: para poder aceder com total conforto e segurança aos dados e aplicações da sua empresa desde o exterior;

- VoIP: para telefonar via Internet;
- Servidor *web*: para disponibilizar conteúdos desde a sua rede local para a Internet.

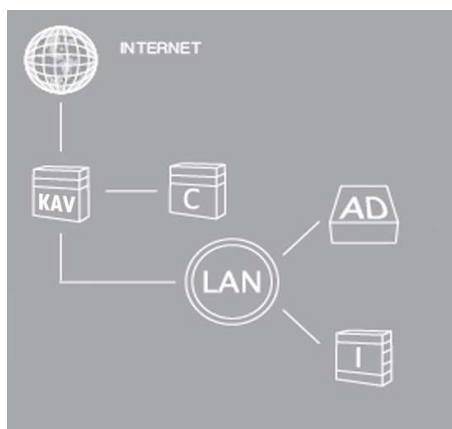


Figura 3.13: IPBrick.KAV - *Appliance* de Segurança [22]

A informação apresentada ao longo desta secção foi recolhida da página oficial do produto [23] e do manual de referência [22].

3.3 Gestix Net

O SGE utilizado para o desenvolvimento desta ferramenta foi o Gestix. O Gestix Net (G-NET) é um *software* modular de vendas e contas correntes de cliente, vendedores, compras e contas correntes de fornecedor e inventário, direccionado para as necessidades das pequenas e médias empresas, incorporado com ERP e CRM (*Customer Relationship Management* - Gestão da Relação com Clientes). Utiliza uma base tecnológica desenvolvida de raiz para fazer face aos novos desafios da actualidade: mobilidade, acessibilidade, segurança, eficiência e versatilidade. O G-NET permite gerir o negócio em qualquer lugar e a qualquer momento, garantindo toda a liberdade e mobilidade através da *web*.

Capítulo 4

Apresentação

4.1 Especificações

A configuração dos modelos de documentos é efectuado graficamente, com a finalidade de facilitar a construção destes. A ferramenta disponibiliza um conjunto de elementos que serão posicionados no modelo da forma que o utilizador pretender. Os elementos são caracterizados por várias propriedades de fácil configuração, assim como disponibiliza uma lista de configurações ao nível da página (p.ex. dimensões, margens, entre outros).

Nas sub-seções seguintes são apresentados os elementos e suas propriedades assim como as configurações e características disponíveis na ferramenta desenvolvida.

4.1.1 Elementos

A ferramenta de criação de templates possui um conjunto de elementos disponíveis, estes permitem efectuar a configuração do aspecto gráfico da página modelo (documento PDF). Os elementos disponíveis são apresentados de seguida.

Texto Fixo

Este elemento permite a introdução de texto no formulário que não irá ser alterado, sendo a informação fixa do documento.

Editor de Texto

Este elemento permitirá ao utilizador criar uma caixa de texto para a edição do mesmo. Este elemento apresenta um menu do tipo ferramenta Office (FCKeditor).

O FCKeditor disponibiliza na *web* muitas funcionalidades dos editores de *desktop*, como o Word. É leve para o sistema e não requer qualquer tipo de instalação no computador do cliente ¹.

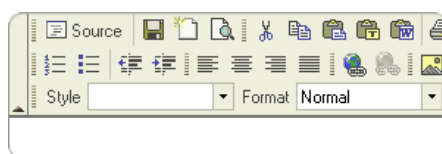


Figura 4.1: Elemento Editor de Texto

Área de Texto

Este elemento permitirá criar uma caixa de texto para edição. Este distingue-se do elemento anterior pelo facto de não possuir qualquer menu de opções.

Box

O elemento box distingue-se do elemento anterior porque a área de texto limita-se a uma linha, sendo útil para pequenos volumes de informação. Com este elemento é possível definir o tipo de informação que deverá ser introduzida, sendo possível definidos os seguintes tipos de dados:

¹Para mais informações sobre este editor consultar a página <http://www.fckeditor.net>

- Numéricos;
- Datas;
- Texto.

Box Dinâmica

Este elemento é idêntico ao interior, contudo, a informação introduzida neste pode ser utilizada para fazer pesquisas na base de dados do ERP, ou seja, o valor deste elemento pode ser utilizada para definir uma condição da pesquisa. Do mesmo modo que acontece no elemento Box, é possível configurar o tipo de dados que este elemento irá receber.

Fórmulas

O elemento fórmula possibilita realização de cálculos matemáticos, utilizando dados dos elementos do tipo Box, Box Dinâmica, ou resultados de outras fórmulas. E além disto, dispõe da possibilidade de utilizar a informação dos elementos do tipo Tabela (ver tópico Tabelas).

Imagem

Esta opção permite colocar imagens sempre que necessário nos documentos que serão produzidos.

Pré-Inserido

Este elemento permite a recolha de dados existentes no gestor de contactos (IP Contactos) de apoio ao sistema de gestão documental, relativamente à entidade associada ao documento. Além da informação da entidade, é possível utilizar a informação de classificação do documento, como por exemplo: responsável pela elaboração do documento, data de elaboração, código ou título do documento, entre outros. Estes tipos de elementos são preenchidos automaticamente no formulário.

Os campos da entidade disponíveis para serem utilizados como campos pré-inseridos foram apresentados anteriormente na tabela 3.1. Na tabela 4.1 são apresentados os campos disponíveis como pré-inseridos com origem no formulário de classificação.

Tabela 4.1: Campos pré-inseridos - Classificação

Pré-inseridos - Documento			
Título	Código	Autor	Assunto
Valor	Centro de Custo	Sub-Centro de Custo	Network
Data de elaboração			

Pop-lists

Os elementos pop-list dividem-se em três tipos:

- Definida: as opções desta pop-list são definidas pelo utilizador;
- Pré-definida: quando introduzido um documento, o utilizador classifica-o através de um formulário. Nesse formulário existe um conjunto de pop-lists para ajudar a fazer a classificação, sendo que este elemento permite a utilização de uma dessas pop-lists. A pop-list disponível contém todos os contactos da entidade a que o documento se refere;
- Dinâmica: este tipo de pop-list é criada dinamicamente através de uma pesquisa à base de dados. Como acontece com a Box Dinâmica, este elemento pode ser utilizado para definir pesquisas na base de dados do ERP. A opção seleccionada deste elemento é utilizada para construir a *query* à base de dados.

O formato da *query* do elemento pop-list dinâmica é o seguinte:

select opcao1, opcao2 from tabela

onde *opcao1* será o código que identifica a descrição, representada por *opcao2*.

A construção deste elemento é efectuada da seguinte forma:

<select>

```
<option value= opcao1 > opcao2 < /option>
```

...

```
<option value= opcao1 > opcao2 < /option>
```

```
< /select>
```

Quando este elemento é utilizado com a finalidade de gerar relatórios dos sistemas ERP é utilizado o value para definir a *query* de pesquisa. A *query* seguinte exemplifica a situação apresentada:

```
select campo1, campo2 from tabela where campo1 = opcao1
```

O resultado da pesquisa à base de dados pode retornar inúmeras opções, tornando este elemento com uma dimensão elevada e de difícil utilização. Para ultrapassar esta situação existe uma opção que, quando seleccionada não serão apresentadas as opções definidas pela pesquisa no formulário. Nesta situação o utilizador efectua a pesquisa no momento do preenchimento do formulário, sendo que a pesquisa pode ser efectuada pela *opcao1* e/ou pela *opcao2*.

Podem existir vários elementos que obsessão às condições da pesquisa, e desta forma o resultado desta será apresentada sobre a forma de uma listagem permitindo ao utilizador seleccionar a opção desejada.

Tabelas

Através deste elemento é possível introduzir tabelas no formulário do modelo do documento, sendo que estão ao dispor do utilizador dois tipos de tabelas: as tabelas de Itens e as Dinâmicas.

- Itens: é possível utilizar este tipo de tabelas, por exemplo, criação de facturas, onde cada linha da tabela corresponde a um serviço ou produto. Sempre que o utilizador preencher e introduzir uma linha, será disponibilizada uma nova linha vazia para preenchimento e assim sucessivamente até ao final. Com este tipo de funcionalidades torna-se necessário efectuar um conjunto de operações matemáticas sobre os dados deste tipo de tabelas, tanto por linhas como por colunas;

- Dinâmica: este tipo de tabelas possibilita produzir relatórios com base na informação contida em bases de dados (ver sub-secção 4.1.4). A informação adquirida será o resultado de uma pesquisa na base de dados de qualquer aplicação. A informação inserida nos elementos Box Dinâmica e na Pop-list Dinâmica, pode ser utilizada para definir as condições de pesquisa à base de dados.

As base de dados suportadas são:

- SQL Server;
- MySql;
- PostgreSQL;
- Oracle.

4.1.2 Propriedades dos Elementos

Depois de uma pequena abordagem aos elementos disponíveis para configuração da página com vista à criação de modelos de documentos, será efectuada uma pequena apresentação das propriedades associadas a estes.

O posicionamento e dimensionamento dos elementos é efectuado de forma gráfica, e estes podem ser dispostos pela página da forma que o utilizador pretender. Para colocar os elementos na página, estes são arrastados do menu correspondente para o local pretendido da página (*drag-drop*). O dimensionamento dos elementos também é efectuado graficamente (*mouse resizable*).

Para garantir um posicionamento gráfico preciso dos elementos, a ferramenta apresenta-o constantemente ao utilizador, ou seja, à medida que um determinado elemento está seleccionado ou está a ser movimentado sobre a tela, a sua posição é constantemente actualizada.

Os elementos apresentam ao utilizador um conjunto de propriedades relativamente

ao nível do texto (p.ex. tamanho de letra, tipo, cor entre outras) Na tabela 4.2 estão apresentadas as propriedades dos elementos.

Tabela 4.2: Propriedades dos Elementos

Posição/Dimensão	Fonte		Espaçamento	
Posição x	Tipo de letra		Espaço entre letras	
Posição y	Estilo da fonte:	Normal	Alinhamento:	Direita
Altura		Negrito		Esquerda
Largura		Itálico		Centro
		Negrito e Itálico		Justo
	Tamanho da letra			
	Cor			
	Efeitos:	Sublinhado		
		Sobrelinhado		
		Riscado		

Estas propriedades são comuns a um grande conjunto dos elementos apresentados anteriormente, sendo aplicadas a todos os elementos em que é possível introduzir texto. Este tipo de propriedades não fazem parte da lista de propriedades do elemento Imagem e Editor de texto, sendo que nestes elementos estão disponíveis apenas as propriedades Posição/Dimensão. As propriedades de cada elemento são alteradas graficamente, sendo que, da mesma forma que o utilizador manipula as opções, pode observar o resultado produzido estando assim o seu trabalho facilitado.

Posicionamento

Durante a configuração do modelo do documento os elementos podem ser posicionados/personalizados como o utilizador pretender, no entanto existe um conjunto de restrições relativas ao posicionamento dos elementos.

Na figura 4.2 estão representadas as restrições definidas relativas ao posicionamento dos elementos.

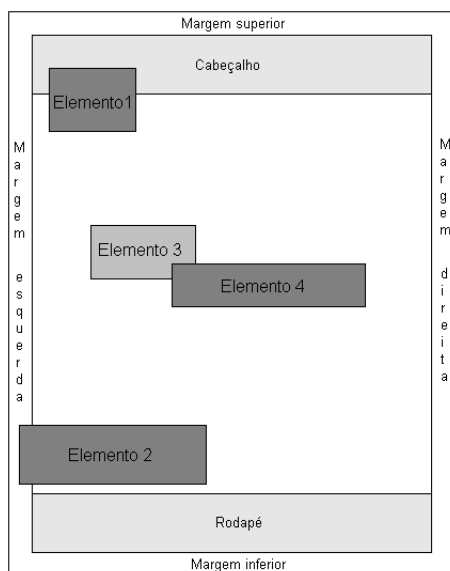


Figura 4.2: Restrições ao nível do posicionamento dos elementos

- Não pode existir elementos sobrepostos (*Elemento3* e *Elemento4*);
- Todos os elementos são colocados na região definidas pelas margens (*Elemento2*);
- Não é possível posicionar elementos simultaneamente em diferentes regiões da página, sendo que, cada elemento ou se encontra do cabeçalho, rodapé ou no corpo da página (*Elemento1*).

Existe um conjunto de elementos que crescem de uma forma não controlada como é a situação da tabela dinâmica (geração de relatórios), não sendo possível saber antecipadamente (no momento da configuração do modelo) o número de linhas da resposta que a pesquisa à base de dados irá retornar. Desta forma o posicionamento dos elementos é relativo permitindo ultrapassar este cenário.

Na existência de elementos deste tipo, o formulário e o documento PDF gerados não apresentam exactamente o mesmo posicionamento configurado no modelo, sendo que os elementos são deslocados verticalmente como está representado na figura 4.3.

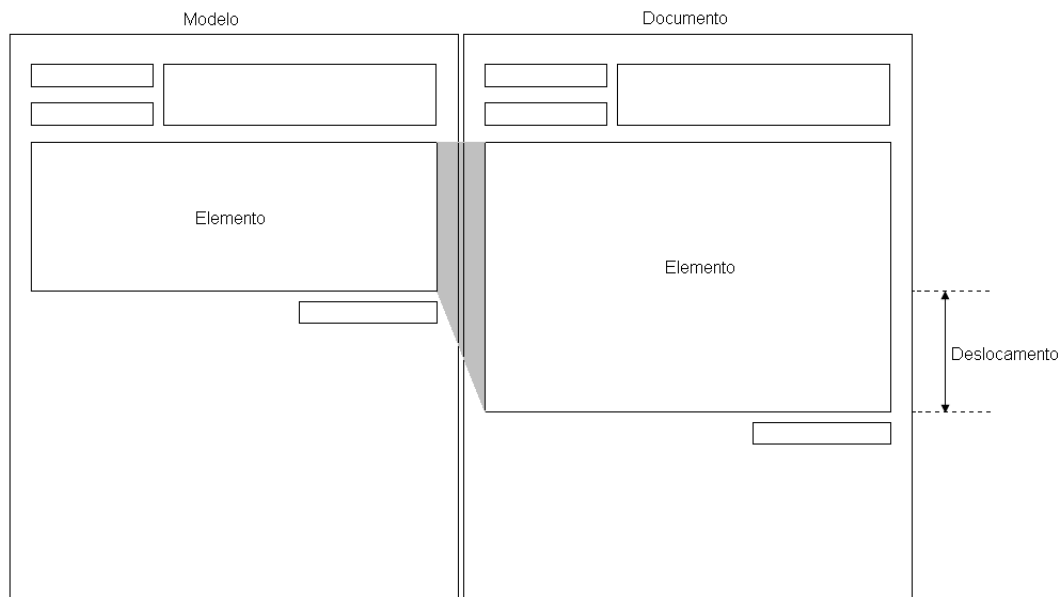


Figura 4.3: Posicionamento relativo dos elementos

4.1.3 Configurações

Página

Esta opção permite configurar a página sobre a qual o modelo do documento é configurado. O utilizador pode, deste modo, seleccionar um tipo de página *standard* (p.ex. A4, A5) onde as dimensões já estão definidas, ou então pode personalizar a própria página. Além da dimensão da página, encontra-se disponível para configuração as quatro margens da mesma, assim como a dimensão do cabeçalho e do rodapé.

Margens

A configuração das margens pode ser efectuada antes de configurar qualquer elemento, ou quando já existirem elementos configurados. As margens, quando configuradas antes de inserir qualquer elemento, têm maior gama de valores que podem tomar. Isto porque, quando não existe qualquer elemento inserido, as margens podem atingir o seu valor máximo, sendo que, caso existam elementos configurados, isto já não acontece.

Para configurar ou alterar os valores das margens quando existe pelo menos um elemento configurado, a configuração vai depender das posições tomadas pelos elementos. A soma das margens superior e inferior depende da distância entre a base do elemento inferior e limite inferior da página. No caso das margens lateral esquerda e lateral direita, dependerá da distância entre a lateral direita do elemento posicionado mais à direita da página e o limite direito da página. Os limites máximos das margens dependerão também das mesmas, caso estejam configuradas. Na figura 4.4 está representada uma página com dois elementos posicionados, onde estão documentadas as dimensões necessárias para que as margens possam ser configuradas.

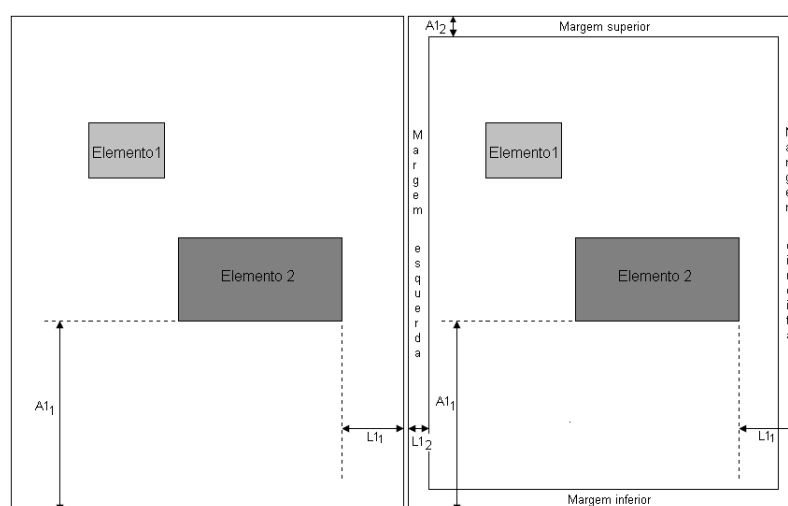


Figura 4.4: Margens da página

Observando a figura 4.4 pode-se verificar que a margem superior e inferior não podem ultrapassar $A1_1 + A1_2$, já $L1_1 + L1_2$ representa o valor máximo que a soma das margens laterais podem ser configuradas. Note-se que a distância entre o limite esquerdo da página e o mesmo limite do *Elemento1* permanecem inalterados, assim como a distância entre o topo da página e o topo do elemento *Elemento1*.

Outra funcionalidade disponível na aplicação é a inclusão de cabeçalho e/ou rodapé nos modelos (ver tópico seguinte), e, neste caso, é necessário ter em questão o espaço disponível na configuração das margens. Como se pode verificar na figura 4.5, para configurar as margens laterais o processo é o mesmo que foi apresentado anteriormente,

$L2_1 + L2_2$. Na presença do rodapé, a soma das margens superiores não podem ultrapassar a distância $A2_1 + A2_2 + A2_3$, correspondente à distância entre a base do elemento inferior (*Elemento2*) e o limite superior do rodapé, acrescentado o valor da margem superior e inferior.

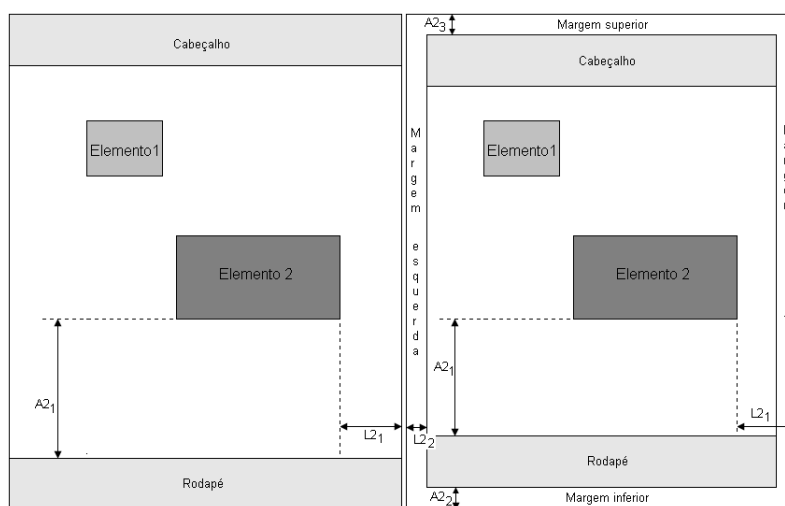


Figura 4.5: Margens da página com cabeçalho e/ou rodapé

Cabeçalho e Rodapé

A ferramenta disponibiliza ao utilizador a possibilidade de introduzir nos modelos de documentos cabeçalho e/ou rodapé, sendo definida altura de cada um. Na região reservada para o cabeçalho e rodapé o utilizador pode configura-los utilizando os elementos anteriormente apresentados.

A altura máxima do cabeçalho e/ou rodapé está dependente das margens verticais (margem superior e inferior) caso existam, e da posição dos elementos se existir algum configurado quando for introduzido o cabeçalho e/ou rodapé. Através da figura 4.6 pode observar-se que o tamanho disponível para o cabeçalho e/ou rodapé é no máximo $A3_1$.

Se já existir cabeçalho e/ou rodapé, o tamanho máximo que estes podem assumir será calculado de forma diferente. Outra situação que interfere com a dimensão do cabeçalho e/ou rodapé é a existência de elementos nestas secções da página, uma vez

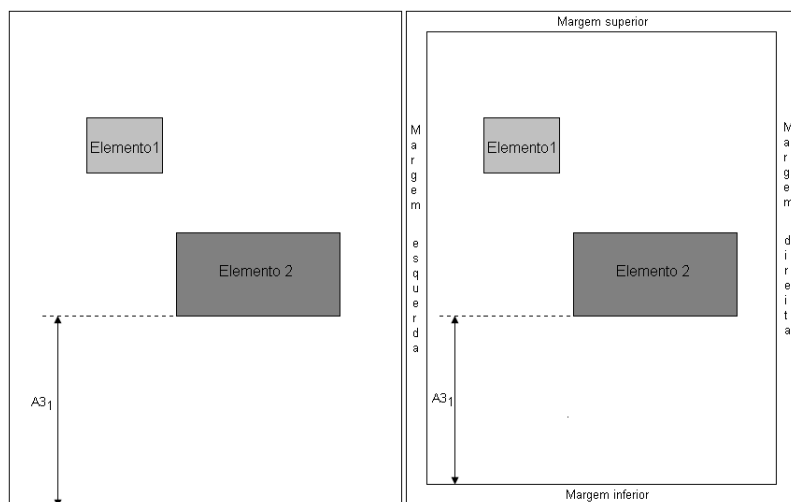


Figura 4.6: Cabeçalho e rodapé

que o limite mínimo é superior a zero.

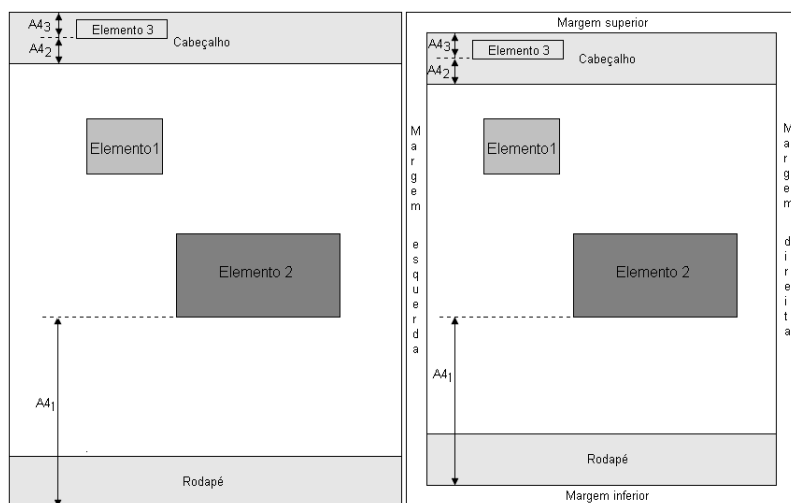


Figura 4.7: Cabeçalho e rodapé com elementos

Através da figura 4.7 pode verificar-se que, a presença de um elemento na secção cabeçalho implica que a dimensão mínima que este poderá ser reconfigurado será de $A4_3$ e, assim, é assegurado que o *Elemento3* fica completo. Pode-se verificar também a implicação no que respeita às dimensões do cabeçalho e/ou rodapé quando este é reconfigurado após uma configuração inicial.

Base de Dados

Esta opção permite configurar a ligação à base de dados do ERP. Para verificar se a configuração está correcta, o utilizador dispõe de uma funcionalidade que verifica a possibilidade de estabelecer a ligação.

Mesmo que a ligação à base de dados não seja possível, os elementos dependentes desta podem ser utilizados no desenvolvimento do modelo. Uma situação comum é o facto dos modelos serem produzidos em servidores diferentes daqueles em que são utilizados, ou seja, é possível exportar o de um servidor e importá-lo para outro.

Assim, torna-se indispensável permitir a utilização dos elementos dependentes de uma ligação à base de dados exterior, mesmo que não seja possível estabelecer esta ligação. Estes elementos apenas estarão disponíveis se existir uma ligação configurada.

Quando o modelo é criado num ambiente onde não é possível estabelecer ligação à base de dados, o utilizador fica sem qualquer *feed-back* dos pedidos que pretende efectuar externamente. A utilização e configuração dos elementos dependentes desta ligação são baseadas apenas na informação fornecida pelo utilizador. Por exemplo: um utilizador que pretende fazer uma pesquisa e comete um erro no nome de um campo. Não existe forma de saber se a *query* possui algum erro deste tipo, apenas é possível detectar erros sintácticos.

Exportação do Documento

A informação introduzida e/ou obtida através do formulário do modelo do documento pode ser exportada. Além da informação ser exportada para PDF, pode também ser exportada para o formato CSV (*comma-separated values*), afim de poder ser interpretada por outras aplicações como por exemplo ferramentas Office.

Importação e Exportação dos Modelos

O desenvolvimento e configuração de modelos de documento pode ser efectuada num servidor para ser utilizado noutro, sendo que esta tarefa é facilitada com a existência da opção de importação e exportação.

4.1.4 Relatórios

A geração de relatórios do ERP é obtida através de pesquisas efectuadas à base de dados. Para a realização da pesquisa deverá ser digitada uma *query* SQL, sendo que o resultado obtido através deste pedido será utilizado para construir o relatório (apenas é permitido efectuar pedidos do tipo *select*). Uma *query* do tipo

*select * from tabela*

não permitirá efectuar qualquer relatório. Uma *query* neste formato permitiria estruturar o relatório, na situação de ser possível estabelecer uma ligação à base de dados durante a configuração do modelo, desta forma, seria possível saber quais os campos disponíveis para configurar a tabela dinâmica.

No entanto, este elemento está disponível quando uma ligação está configurada como foi referido. Mesmo que, no momento do desenvolvimento do modelo do documento, não seja possível estabelecer uma ligação à base de dados é possível configurar o relatório, mas para isso deverão estar definidos todos os campos pretendidos. De seguida, está representado um exemplo de uma *query*, que pode ser utilizada para produzir um relatório:

select campo1, campo2 from tabela

onde *campo1* e *campo2* são os campos pretendidos para gerar o relatório.

Através da *query* apresentada, é possível realizar uma simples pesquisa sobre uma tabela, mas também é possível desenvolver *queries* complexas onde estão envolvidas várias tabelas, ordenações (*order by*) entre outros. No entanto é necessário que sejam

especificados todos os campos pretendidos como é o caso seguinte:

```
select tabela1.opcao1, tabela1.opcao2, tabela2.opcao2 from tabela1, tabela2
```

```
where tabela1.opcao1 = tabela2.opcao1 order by tabela1.opcao2
```

Quando é configurado um relatório sem a possibilidade de estabelecer uma ligação à base de dados, o utilizador não possui qualquer *feed-back* sobre a validade da *query* desenvolvida, nesta situação será efectuada uma validação de sintaxe da *query*. Se enquanto o template estiver a ser desenvolvido houver a possibilidade de existir uma ligação activa, o utilizador será informado de qualquer problema que possa existir.

Relatórios Utilizando Informação do Formulário

Até este momento foi apresentado o processo de configuração que permite efectuar pesquisas na base de dados do ERP, onde o utilizador não tem qualquer campo para personalizar a pesquisa, tornando os relatório pouco úteis. Para responder à necessidade de produzir relatórios restritos às condições definidas pelo utilizador, são apresentados ao mesmo um conjunto de elementos que podem ser utilizados para definir os critérios de pesquisa (construção da *query*), denominados de elementos dinâmicos (box dinâmica e pop-list dinâmica).

Na figura 4.8 está esquematizado o processo de geração de relatórios através de informação fornecida pelo utilizador. Verifica-se que existe um conjunto de elementos do formulário que são utilizados para definir as condições da pesquisa (*query*). Com o resultado da pesquisa são efectuados as operações definidas pelo utilizador e de seguida será apresentado o relatório.

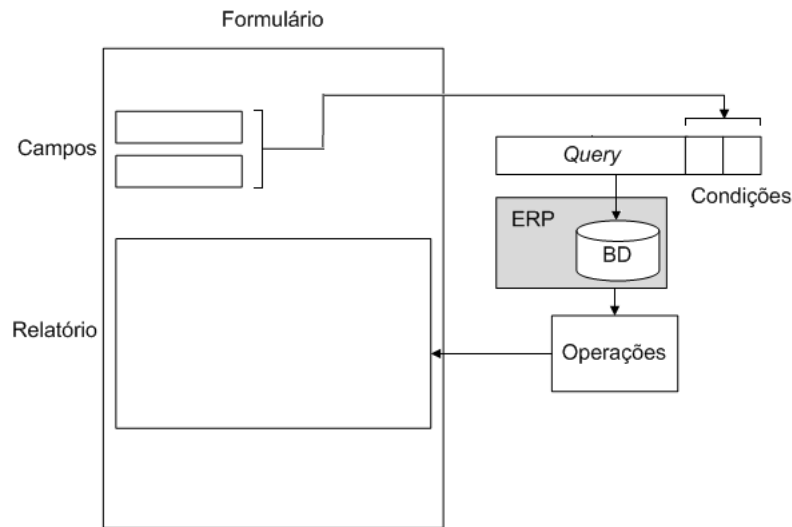


Figura 4.8: Geração de relatórios

4.2 Design da Ferramenta

4.2.1 Ficheiros Gerados

Após a configuração do modelo do documento será gerado um conjunto de ficheiros que permitem manipular toda a informação do documento assim como efectua a integração com os diversos sistemas anteriormente apresentados. Todos os ficheiros produzidos no processo de geração são armazenados num armazém dedicado para o efeito, este é organizado por directorias, tantas quanto o número de modelos gerados e no interior destas directorias estão os ficheiros correspondentes a cada modelo.

Após a geração do modelo configurado, esta ferramenta de desenvolvimento não será necessária para o restante funcionamento do sistema, o funcionamento é garantido pelos ficheiros gerados.

Todos os ficheiros são gerados com base na informação de configuração dos modelos, a criação dos ficheiros é efectuada de forma dinâmica. Os ficheiros gerados são designados da seguinte forma:

- modelo.php;

- modelo.xml;
- modeloAccao.php;
- modeloFormulario1.xml, modeloFormulario2.xml, modeloFormulario3.xml;
- modelo.class;
- modeloDB.class;
- modeloPesquisa.php;
- modeloXML.php;

modelo.php

O ficheiro modelo.php é o primeiro a ser utilizado, sendo utilizado no processo de criação/revisão de documentos. Este documento permite fazer a validação do formulário de classificação, reunir e preparar toda a informação necessária para o formulário do modelo a apresentar ao utilizador. Com a informação reunida é criado um documento XML sendo aplicada uma transformação XSTL que permite apresentar o formulário ao utilizador.

modeloAccao.php

Este ficheiro permite fazer a validação dos dados introduzidos pelo utilizador no formulário apresentado. Neste ficheiro é executada a criação do documento PDF, utilizando o ficheiro XML onde se encontra armazenada toda a informação, e, a este, será aplicado o formato da página anteriormente definido (ficheiro modelo.xml). Neste ficheiro também é efectuada todas as operações sobre a base de dados do SGDW e o armazenamento da informação no sistema de ficheiros.

modelo.class

Neste ficheiro estão definidas um conjunto de classes responsáveis pela manipulação e gestão a informação introduzida e/ou gerada no formulário no cenário de geração de relatórios.

modeloDB.class

O ficheiro modeloDB.class representa a camada de acesso a dados, permitindo estabelecer e gerir a ligação à base de dados bem como efectuar pesquisas sobre a mesma. Este ficheiro é criado se existir uma ligação à base de dados configurada.

modeloPesquisa.php

Este ficheiro é criado se existir configurado pelo menos um elemento do tipo pop-list com a opção de pesquisa activa. Este documento permite ultrapassar as dificuldades que ocorrem quando as opções dos elementos pop-list é elevado, que através de um pequeno formulário permite ao utilizador pesquisar a opção pretendida. Desta forma, o utilizador introduz a informação que pretende pesquisar, sendo retornado as opções que obedecem à pesquisa.

modeloXML.php

Este ficheiro permite fazer a exportação da informação do documento para o formato csv.

Ficheiros .xsl

Os ficheiros .xsl representam o estilo como é apresentada a informação ao utilizador. O ficheiro modelo.xsl representa o estilo do documento PDF, e os restantes definem o estilo de apresentação da informação sobre o formato de HTML (formulário).

De acordo com o cenário em que o utilizador preenche o formulário do modelo será utilizado um estilo diferente. O formulário pode ser apresentado em três situações distintas: introdução, revisão e revisão por uma acção definida no *workflow*. Desta forma em cada uma das situações será utilizado o estilo XSL correspondente.

4.2.2 Criação de Documentos

A criação do documento passa por várias fases como pode ser analisado através da figura 4.9. O processo começa pela classificação do documento a ser produzido, como foi apresentado anteriormente, seguido por um processo de validação desta informação. A informação de classificação é armazenado num documento XML (XML Formulário), e a esse documento é aplicado um estilo que permite apresentar a informação do XML (transformação XSLT), ou seja, esta transformação gera o formulário do modelo do documento permitindo a introdução da informação.

No formulário encontram-se preenchidos os elementos cuja informação tem origem no formulário de classificação, sendo os restantes preenchidos pelo utilizador. Se o utilizador tiver configurado elemento(s) que permite fazer uma pesquisa à base de dados do ERP, este pode submeter uma pesquisa à base de dados e, com a informação de retorno, é gerado o relatório desejado.

Quando é efectuado a submissão do formulário é executado um processo de validação da informação e caso a informação seja válida é criado um novo ficheiro XML (XML Template). A este documento XML é aplicado um estilo que permite criar um documento PDF através de uma transformação.

4.2.3 Informação XML

A informação dos documentos definidos por esta ferramenta é gerida e armazenada no formato XML. Os ficheiros XML são apresentados ao longo desta sub-secção.

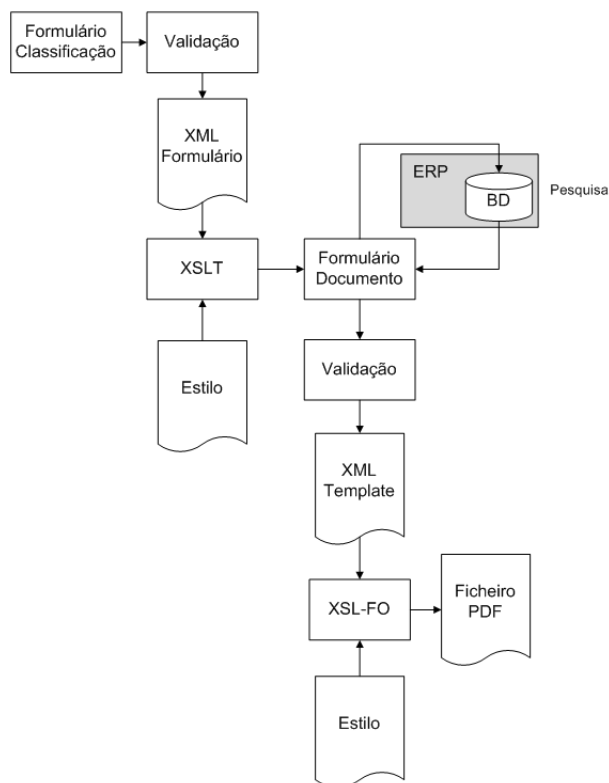


Figura 4.9: Criação de documentos PDF

Ficheiro XML Formulário

O ficheiro XML é gerado dinamicamente de acordo com a configuração do modelo do documento. A este ficheiro XML será aplicado um estilo que, através de uma transformação XSLT, apresenta o conteúdo deste no formulário do modelo do documento.

A informação que compõe o ficheiro XML pode ter várias origens:

- Formulário de classificação do documento;
- Base de dados do SGDW;
- Base de dados do ERP;
- Ficheiro XML Documento - este ficheiro é utilizado quando se efectua uma revisão do documento.

Através da figura 4.10 está representado o diagrama ilustrativo do processo de criação do ficheiro XML Formulário. Para classificar um documento, o utilizador responde a um formulário de classificação, sendo que, quando o modelo do documento é configurado para utilizar a informação da classificação, esta será apresentada no formulário sem que o utilizador a digite novamente.

Quando são utilizados elementos pré-inseridos relativos à informação da entidade que o documento deverá estar associado, é necessário recolher a informação da base de dados, pois esta informação não vem directamente do formulário de classificação do documento.

A informação deste documento pode ter origem na base de dados de uma aplicação exterior, isto verifica-se quando é gerado um relatório estatístico.

Por fim, a informação pode estar armazenada num outro ficheiro XML caso o utilizador pretenda efectuar a revisão de um documento já existente no SGDW. Nesse ficheiro encontra-se toda a informação introduzida no formulário do modelo aquando da introdução no SGDW.

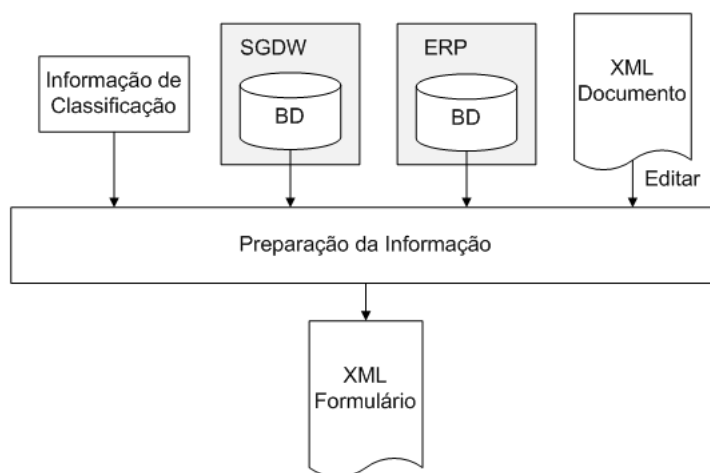


Figura 4.10: Criação dos ficheiros XML Formulário

Ficheiro XML Documento

A informação introduzida no formulário do modelo do documento é armazenado num ficheiro no formato XML. Este ficheiro é utilizado para produzir o ficheiro PDF, a este ficheiro é aplicado um estilo que, através de uma transformação produz o documento PDF.

```
<?xml version="1.0" encoding="UTF-8" ?>
<template>
  <var1>Relatório/var1>
  <var2>2008-02-28</var2>
  <var3>
    <n_item>1</n_item>
    <var3_1>10</var3_1>
    <var3_2>20</var3_2>
    <n_item>2</n_item>
    <var3_1>20<var3_1>
    <var3_2>40<var3_2>
  </var3>
  <formula1>100</formula1>
</template>
```

Figura 4.11: Ficheiro XML Documento

No exemplo apresentado na figura 4.11 estão disponíveis quatro elementos, representados pelos nós var1, var2, var3, formula1. Note-se que em todos os nós do ficheiro XML são terminados por um número, sendo que este número é gerido pelo sistema. Este identificador nunca será repetido em ficheiros XML de formulários diferentes, ou seja, o formulário A produz um ficheiro com o nó var1, logo nos ficheiros do formulário B não existe nenhum nó var1.

Os nós var1 e var2 podem representar o conteúdo dos elementos box, box dinâmica ou área de texto. Podem representar também a opção seleccionada da lista de opções de um elemento pop-list definida, dinâmica ou pré-definida.

O nó var3 representa um elemento do tipo tabela (Itens ou Dinâmica), e essa tabela possui duas colunas representadas pelo nó var3_1 e var3_2. As tabelas podem ter várias linhas, sendo que essas linhas são identificadas pelo nó item, logo, cada linha é constituída pela coluna var3_1 e var3_2 como se pode verificar no exemplo anterior.

Para elementos do tipo fórmula o seu resultado é apresentado de forma idêntica ao nó `formula1`.

4.2.4 Armazenamento da Informação

A informação introduzida no formulário de apoio à criação do documento PDF é armazenada em ficheiros XML, e estes ficheiros estão armazenados num armazém dedicado. Como foi apresentado, esta ferramenta funciona através da informação armazenada em ficheiros XML, e essa informação é apresentada através de transformações XSL que permitem apresentar a mesma através dos estilos (estilo do formulário e formato do PDF) definidos pelo utilizador quando faz a configuração do formulário.

No caso de ter sido seleccionado a opção estatísticas no momento da criação do modelo, a informação além de ser armazenada em XML será armazenada na base de dados. Quando é efectuada a geração dos ficheiros anteriormente apresentados, são criadas tabelas na base de dados necessárias à gestão da informação introduzida através dos formulários. Por cada tabela (itens ou dinâmica) é criada uma tabela com todos os campos da respectiva tabela e será preenchida com a informação que o utilizador introduziu através da tabela de itens, ou então com a informação recebida através da *query* executada à base de dados no caso da tabela dinâmica. A restante informação do formulário será armazenada numa outra tabela, com excepção dos elementos imagem, texto fixo e elementos do tipo tabela (itens e dinâmica) que serão armazenados separadamente como acabou de ser analisado.

Os documentos PDF criados através desta ferramenta são armazenados no sistema de ficheiros do SGDW, seguindo o seu fluxo de trabalho.

Além das formas apresentadas de armazenar a informação, a ferramenta disponibiliza um módulo para exportação dos dados do documento, sendo estes por sua vez exportados para um ficheiro no formato CSV.

Armazéns

O armazenamento da informação é efectuada através de armazéns específicos para cada formato de dados, desta forma, são utilizados três ou quatro armazéns dependendo da configuração do utilizador:

- Armazém de imagens;
- Armazém XML;
- Armazém PDF;
- Base de dados (quando a opção estatísticas estiver activa).

O armazém de imagens é utilizado durante a configuração do modelo do documento e, sempre que o utilizador introduzir uma imagem no modelo, esta será armazenada neste armazém. As imagens deste armazém são utilizadas para apresentar no formulário do documento e no respectivo documento PDF.

Quando um documento é criado ou é efectuada uma revisão são utilizados os restantes armazéns referenciados, sendo que a informação será armazenada no formato XML (armazém XML), no sistema de ficheiros do SGDW (armazém PDF) e, por fim, será armazenada na base de dados. Quando um documento for eliminado, será eliminado dos três armazéns (ver figura 4.12).

4.2.5 Modelo de Dados

O modelo entidade-relação representado na figura 4.13 define as entidades e respectivos atributos de parte da base de dados para o desenvolvimento da ferramenta de apoio à criação de modelos de documentos e integração com o SGDW.

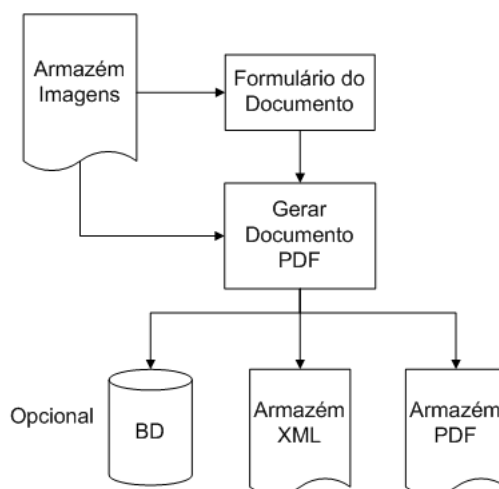


Figura 4.12: Armazéns da aplicação

4.2.6 Modelo de Camadas

O desenvolvimento da aplicação seguiu uma arquitectura em três camadas (ver figura 4.14). A arquitectura em três camadas, envolve a separação das funcionalidades usando estratos, com o objectivo de separar a lógica de apresentação, a lógica de negócio e a ligação com a base de dados (lógica de acesso a dados). A separação em três camadas torna o sistema mais flexível, de modo que partes podem ser alteradas independentemente. Com o emprego de arquitectura em três camadas, qualquer alteração numa determinada camada não influi nas demais, desde que o mecanismo de comunicação entre elas permaneça inalterado.

Acesso a Dados

É uma camada separada da camada da apresentação de dados que tem a função de efectuar a conexão com a fonte de dados e através de comandos SQL manipularlos. Esta separação facilita a manutenção e a flexibilidade da aplicação. A camada de apresentação, quando deseja exibir algum dado, chama a camada de acesso aos dados e solicita a informação. A camada de apresentação não acede a dados nem a camada de dados faz apresentação.

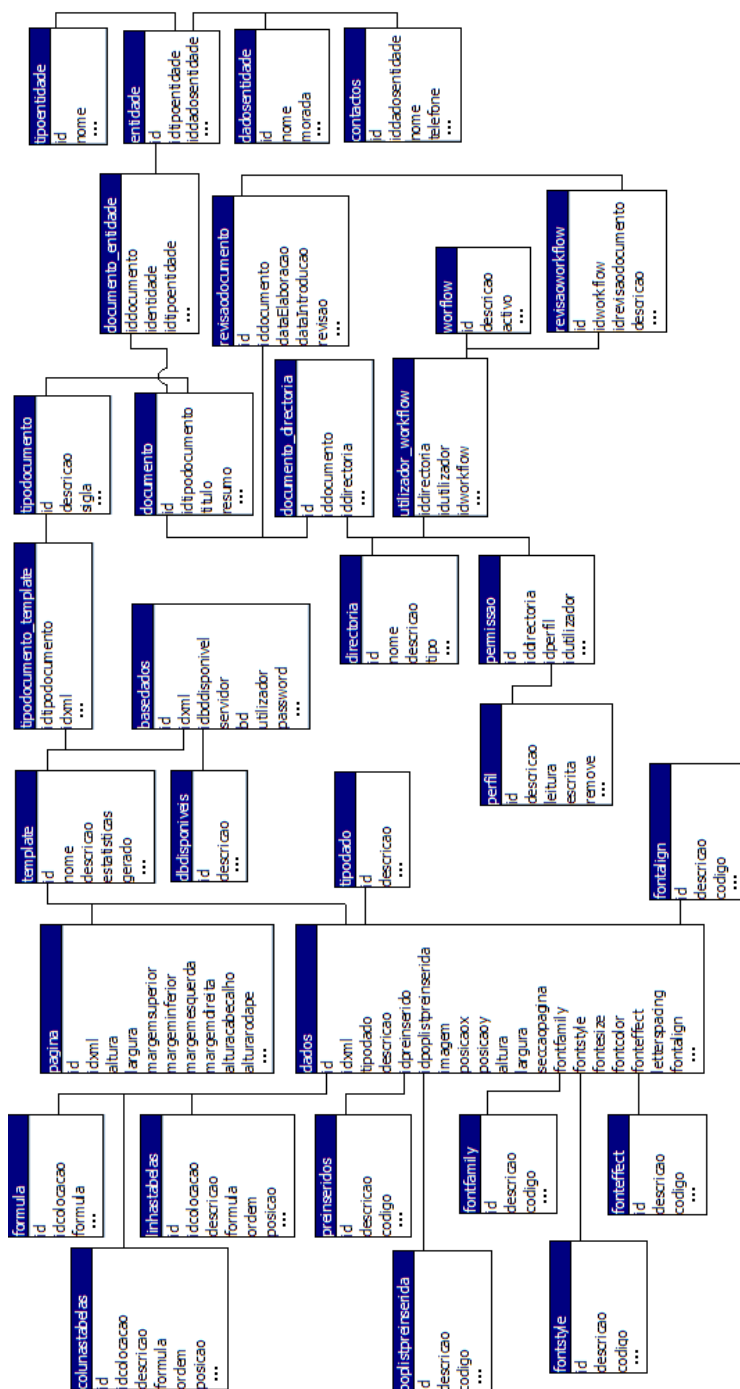


Figura 4.13: Modelo de dados

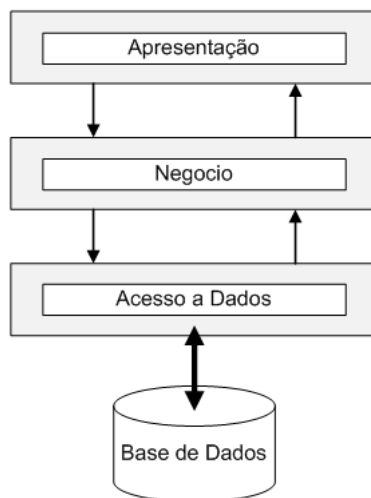


Figura 4.14: Arquitectura três camadas

A camada de acesso a dados está organizada por várias classes, que permitem manipular a informação da base de dados. Estas classes possibilitam:

- Criar registos na base de dados;
- Ler registos na base de dados e retornar os dados;
- Actualizar registos no base de dados;
- Excluir registos da base de dados.

A camada de acesso a dados foi desenvolvida em programação orientada a objectos. São utilizadas um conjunto de classes que permitem manipular a informação da base de dados, e existe uma outra classe que responsável por gerir as ligações à base de dados.

4.2.7 Assunções

Nesta secção estão apresentadas algumas considerações tomadas para o desenvolvimento da ferramenta.

A possibilidade desta ferramenta gerar relatórios com base na informação armazenada na base de dados do ERP, implica que seja necessário efectuar pesquisas sobre as mesmas. Para construir as *queries* que permitem efectuar a consultas, assume-se que o utilizador tenha conhecimentos sobre SQL.

Assumiu-se também que todos os elementos que dependem dos resultados das pesquisas efectuadas à base de dados do ERP podem ser utilizados, mesmo que não seja possível estabelecer uma ligação à base de dados. Muitas vezes, os templates gerados por esta ferramenta são criados fora do ambiente em que irão funcionar e, desta forma, não é possível estabelecer a ligação. Deste modo, para utilizar este tipo de elementos, o requisito é ter uma ligação configurada mesmo que esta não se possa estabelecer.

4.3 Implementação

Nesta secção será efectuada uma análise e apresentação das funcionalidades disponíveis pela ferramenta. As funcionalidades referenciadas ao longo desta secção estão disponíveis apenas para utilizadores com permissões para desenvolver modelos de documentos.

A interface está dividida em três partes que permitem a interacção com o utilizador, como está representado na figura 4.15:

1. Menus;
2. Área de configuração do modelo do documento;
3. Área de informação do posicionamento dos elementos.

Os elementos e configurações disponíveis nesta ferramenta estão divididos por cinco menus designados da seguinte forma:

- Menu Campos;
- Menu Tabelas;

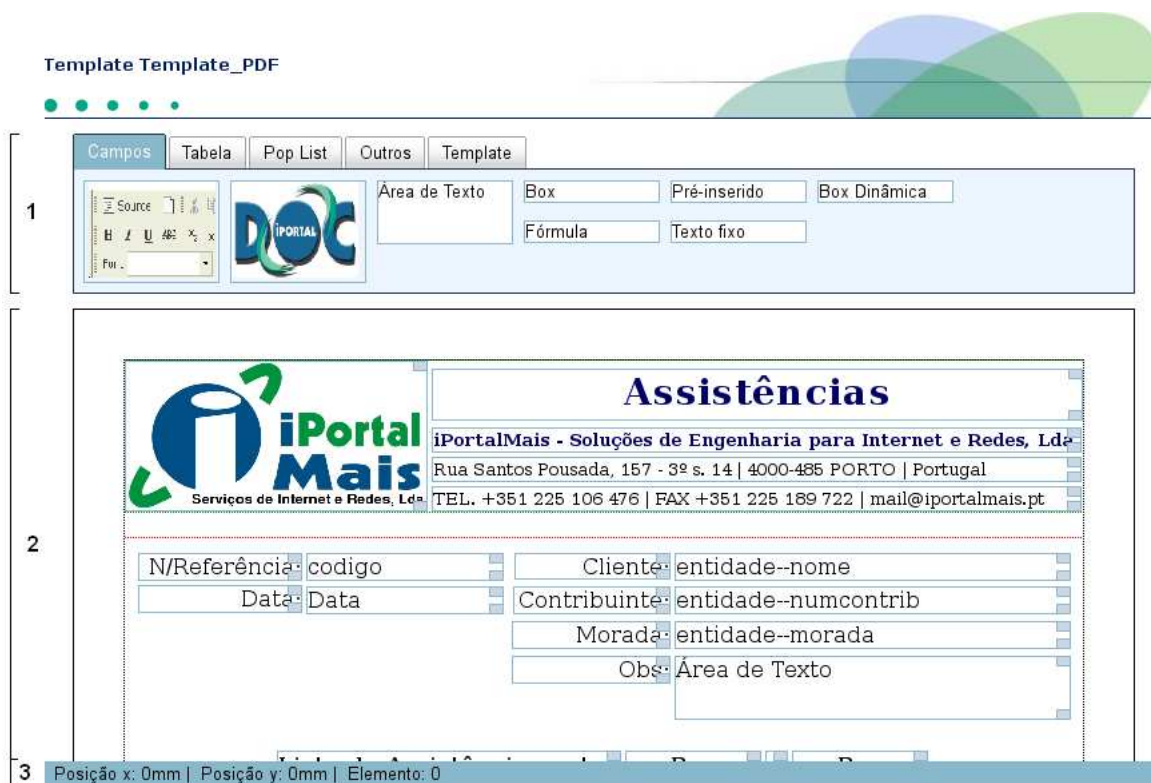


Figura 4.15: Áreas de navegação

- Menu *Pop-lists*;
- Menu Outros;
- Menu Templates.

A apresentação dos menus referidos é descrita nas próximas sub-seções.

4.3.1 Menu Campos

No menu Campos estão disponíveis o seguinte conjunto de elementos:

- Editor de Texto;
- Box;

- Box dinâmica;
- Área de Texto;
- Fórmula;
- Imagem;
- Pré-Inserido;
- Texto Fixo.

O elemento box dinâmica está disponível apenas quando existir uma ligação à base de dados configurada, uma vez que este elemento é utilizado para definir uma condição de pesquisa permitindo construir relatórios com base na informação dos sistemas ERP.

Na figura 4.16 está representado o menu Campos.

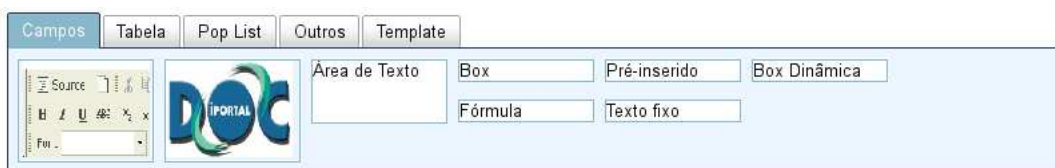


Figura 4.16: Elementos do menu Campos

4.3.2 Menu Tabelas

No menu Tabelas estão disponíveis os dois tipos de tabelas:

- Tabela de Itens;
- Tabela Dinâmica - utilizado para gerar relatórios (ver sub-secção 4.1.4)

O elemento tabela dinâmica é utilizada para apresentar os resultados obtidos dos sistemas ERP, e desta forma estará disponível na existência de uma ligação à base de dados configurada.

O menu Tabelas esta representado na figura 4.17.

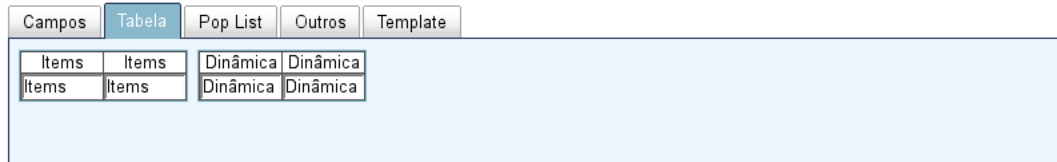


Figura 4.17: Elementos do menu Tabelas

4.3.3 Menu Pop-list

Neste menu, encontram-se disponíveis os vários elementos do tipo pop-list:

- Definida;
- Pré-definida;
- Dinâmica.

Na figura 4.18 é apresentado o menu pop-list.



Figura 4.18: Elementos do menu Pop-list

As pop-lists definida e dinâmica são configuradas como está representado nas figuras 4.19 e 4.20. Para definir as opções da pop-list definida é introduzida uma opção por cada linha como está representado (opções: Sim ou Não). No caso da pop-list dinâmica as opções são definidas pela *query* definida.

O elemento pop-list dinâmica está disponível apenas quando existir um ligação à base de dados configurada. A opção seleccionada deste elementos pode ser utilizada para

Figura 4.19: Pop-list definida

Figura 4.20: Pop-list dinâmica

definir uma condição de pesquisa de forma a construir relatórios com base na informação dos sistemas ERP.

Quando é utilizado este elemento com a opção de pesquisa configurada existirá um link permitindo ao utilizador aceder ao formulário de Pesquisa. Na figura 4.21 está representado um exemplo de um formulário de pesquisa. Pode verificar-se que o formulário dois campos sobre os quais o utilizador pode efectuar a pesquisa, a descrição e o id da opção.

Figura 4.21: Elemento pop-list dinâmica - Formulário de pesquisa

4.3.4 Menu Outros

Neste menu encontram-se disponíveis as configurações da página e da ligação à base de dados.

Página

A opção página oferece ao utilizador um conjunto de propriedades para formatação da página:

- Dimensão (opção representada na figura 4.22);
- Margens (opção representada na figura 4.23);
- Cabeçalho e Rodapé (opção representada na figura 4.24).

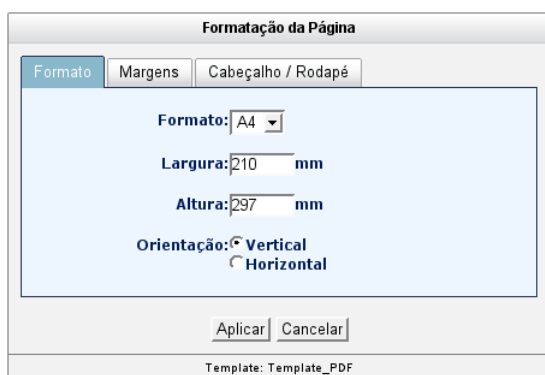


Figura 4.22: Menu formato da página

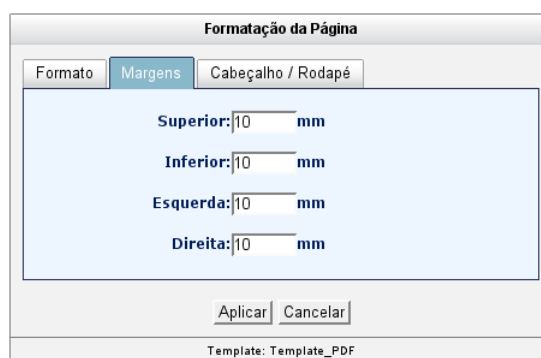


Figura 4.23: Menu margens da página

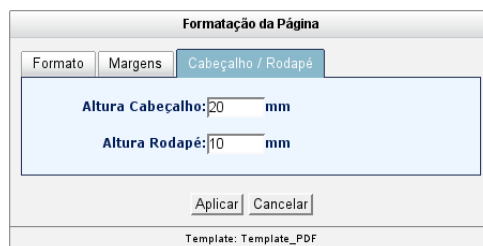


Figura 4.24: Menu cabeçalho / rodapé da página

Base de Dados

A opção Base de Dados permite configurar a ligação à base de dados. Na figura 4.25 está representado o formulário que permite a configurar a ligação. Com o objectivo de garantir que a configuração está correcta, o utilizador tem ao seu dispor uma funcionalidade que permite testar a ligação.

O formulário, intitulado "Configurar Ligação à BD", possui uma aba "Configuração". Contém os seguintes campos e botões:

- Tipo:** Menu suspenso com "PostgreSQL" selecionado.
- Nome do Servidor:** Campo de texto com o valor "localhost".
- Porta do Servidor:** Campo de texto com o valor "5432".
- Nome da DB:** Campo de texto com o valor "database".
- Utilizador da DB:** Campo de texto com o valor "user".
- Palavra-passe da BD:** Campo de texto com caracteres ocultos por pontos.
- Testa ligação:** Botão para verificar a conexão.
- Alterar / Cancelar:** Botões para salvar ou cancelar as alterações.
- Template: Template_PDF:** Texto no rodapé do formulário.

Figura 4.25: Configuração da base dados externa

4.3.5 Menu Template

Para finalizar, o menu Template que dispõe das seguintes opções:

- Gerar;
- Guardar como;
- Exportar;
- Alteração das Formatações;
- Pré-visualização do Formulário;

- Pré-visualização do Documento.

Na figura 4.26 está representado o menu Template.



Figura 4.26: Menu Template

4.3.6 Propriedades dos Elementos

As propriedades dos elementos já foram apresentadas anteriormente na secção 4.1.2. Nesta secção, é apresentado o ambiente que o utilizador encontra no momento da configuração dos elementos.

Cada elemento possui dois botões que permite ao utilizador:

1. Dimensionar ou Re-dimensionar o elemento;
2. Aceder às propriedades do elemento.

As propriedades dos elementos estão divididas por menus:

- Posição / Dimensão (propriedade representada na figura 4.28);
- Fonte (propriedade representada na figura 4.30);
- Alinhamento (propriedade representada figura 4.29).

Estes menus estão disponíveis em quase todos os elementos apresentados, e além destes menus cada elemento possui outras opções que permitem caracterizar o próprio elemento.

Template Template_PDF

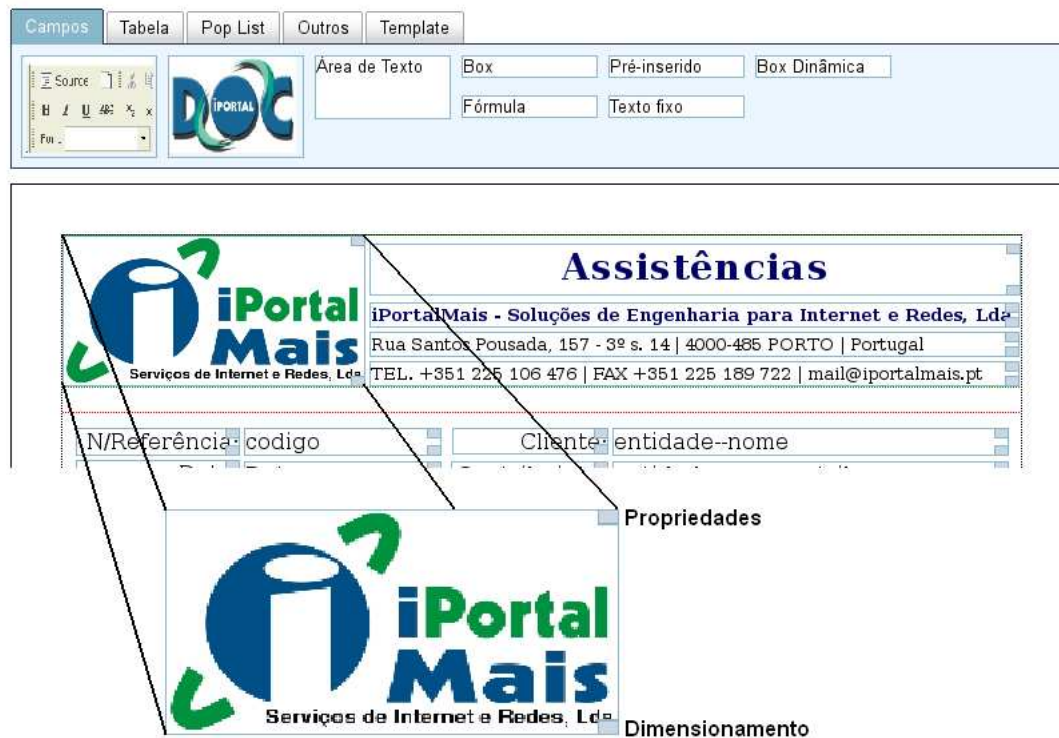


Figura 4.27: Opções dos elementos



Figura 4.28: Menu Posição / Dimensão



Figura 4.29: Menu Alinhamento

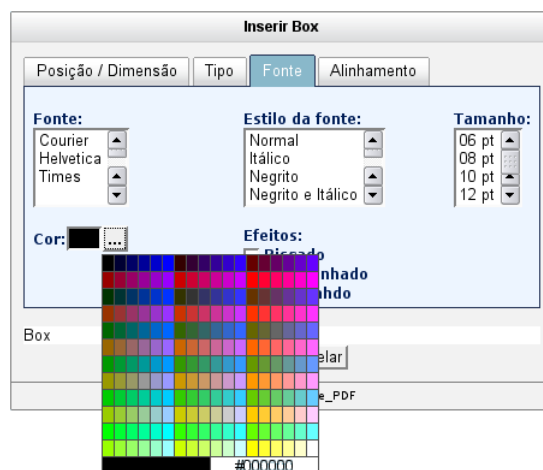


Figura 4.30: Menu Fonte

4.3.7 Utilização dos Modelos

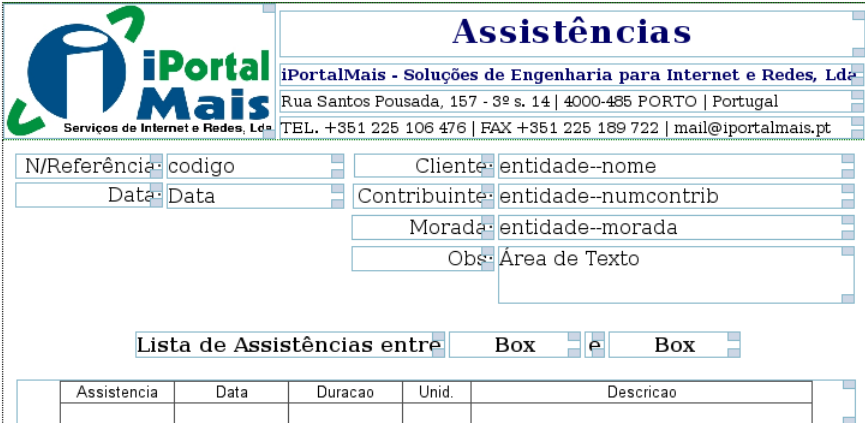
Ao longo da secção foi apresentada de uma forma geral as funcionalidades que permitem o utilizador desenvolver modelos de documentos. Agora é apresentada a ferramenta na perspectiva da utilização dos modelos anteriormente configurados.

Na perspectiva de utilização dos modelos, os ficheiros criados aquando da criação dos modelos são utilizados permitindo fazer a integração entre os diversos sistemas.

Para demonstrar a utilização dos modelos apresenta-se um modelo (ver figura 4.31) com o objectivo de gerar relatórios das assistências efectuados sobre uma entidade durante um período de tempo definido por duas datas.

O formulário correspondente ao modelo apresentado está representado na figura 4.32. No formulário pode observar-se o relatório de assistências relativas à entidade Rui Carvalho entre as datas 2008 – 02 – 27 e 2008 – 02 – 29.

Após a submissão do formulário, este fica disponível no SGDW como está representado na figura 4.33.



iPortalMais
Serviços de Internet e Redes, Lda

Assistências

iPortalMais - Soluções de Engenharia para Internet e Redes, Lda
Rua Santos Pousada, 157 - 3º s. 14 | 4000-485 PORTO | Portugal
TEL. +351 225 106 476 | FAX +351 225 189 722 | mail@iportalmais.pt

N/Referência: código Cliente: entidade--nome
Data: Data Contribuinte: entidade--numcontrib
Morada: entidade--morada
Obs: Área de Texto

Lista de Assistências entre e

Assistencia	Data	Duracao	Unid.	Descricao

Figura 4.31: Modelo de documento



iPortalMais
Serviços de Internet e Redes, Lda

Assistências

iPortalMais - Soluções de Engenharia para Internet e Redes, Lda
Rua Santos Pousada, 157 - 3º s. 14 | 4000-485 PORTO | Portugal
TEL. +351 225 106 476 | FAX +351 225 189 722 | mail@iportalmais.pt

N/Referência: TP 1/2008 Cliente: Rui Carvalho
Data: 28/2/2008 Contribuinte: 123456789
Morada: Rua da Cachada nº103
4750-726 Barcelos
Portugal
Obs:

Lista de Assistências entre 2008-01-27 e 2008-03-29

Assistencia	Data	Duracao	Unid.	Descricao
50396	2008-02-28	1.0000	Horas	Serviços iPortaldoc
50397	2008-02-28	10.0000	Horas	Serviços IPBrick
50398	2008-02-28	3.0000		Serviços iPortaldoc

Figura 4.32: Formulário de documento

1 - 1 Total 1 documentos		Ver Sumários
	Documento Template PDF 1	 +info Desc pdf

Figura 4.33: Lista de documentos

Capítulo 5

Conclusão

Neste capítulo é feita uma síntese do trabalho elaborado, realçando-se os aspectos de maior interesse.

A importância da informação para as organizações é universalmente aceite, constituindo, um dos recursos mais importantes cuja gestão e aproveitamento estão directamente relacionados com o sucesso. A informação também é considerada e utilizada em muitas organizações como uma estrutura e um instrumento de gestão. Os SGD são uma ferramenta poderosa permitindo fazer o armazenamento e gestão de toda a informação das organizações.

A utilização dos SGW permitem automatizar o processo de negócio das organizações. A execução de processos de negócio de sistemas empresariais é importante e significativa na medida em que, a partir de um modelo de processos de negócio de uma empresa podem gerar automaticamente um conjunto de tarefas bem definidas. Através dos SGW é possível automatizar algumas destas tarefas, permitindo também a monitorização do fluxo de informação ou do estado do processo de negócio.

As empresas recebem e produzem uma enorme quantidade e tipo de informação. A organização e tratamento de toda a informação é muito importante no mundo empresarial, sendo que os SGD e os SGW são um poderoso auxiliar para qualquer empresa. Desta forma, a interligação destes com outras aplicações torna-se um passo importante e

inevitável no desenvolvimento das empresas.

Quase todas as empresas possuem aplicações que não comunicam entre si e que foram desenvolvidas usando tecnologias, linguagens de programação, estruturas de base de dados e plataformas diferentes. Devido a estes factores adversos, torna-se muito difícil integrar as diferentes aplicações das empresas. No entanto, a integração das várias aplicações é um passo muito importante na vida das empresas, permitindo obter um verdadeiro sistema empresarial e, desta forma, aumentar o seu rendimento e apresentar-se como uma empresa realmente competitiva.

Através da ferramenta apresentada é possível um funcionamento integrado do SGDW com os sistemas ERP, possibilitando recolher informação destes sendo gerida e controlada pelo SGDW, permitindo ganhos elevados no tratamento da informação. Este modelo de integração permite o fluxo de informação num único sentido, dos sistemas ERP para o SGDW.

Os sistemas ERP apresentam-se como soluções que, de uma forma geral, tendem a adaptar-se a qualquer empresa, enquanto que estes, dispõem de um conjunto de relatórios que podem ou não, ir de encontro às necessidades da organização. Com esta ferramenta é possível ultrapassar as dificuldades que as organizações sentem para produzir relatórios específicos do seu sistema ERP.

Com a implementação desta solução verifica-se uma redução dos custos associados ao tratamento e criação de documentos. Através de um sistema tradicional os utilizadores necessitam de utilizar uma ferramenta para formatação do documento (p.ex. ferramentas Office) e posteriormente utilizar o SGDW para armazenar e controlar o ciclo de vida do documento. Com esta ferramenta este cenário deixa de existir, uma vez que o utilizador responde apenas a um formulário para obter o resultado desejado, um documento no SGDW.

Desta forma as empresas não sentem a necessidade de ter disponível nos seus postos de trabalho ferramentas de apoio à criação de documentos, uma vez que o SGDW e a respectiva ferramenta está embebida no sistema de comunicações, sendo necessário apenas um navegador *web* para aceder ao SGDW.

Com esta solução é eliminado o risco de existirem documentos do mesmo tipo personalizados de forma diferente. Por exemplo: com ferramentas do tipo Office o utilizador tem a liberdade para formatar os documentos, no entanto esta ferramenta permite ultrapassar esta situação, uma vez que a formatação do documento não está disponível para todos os utilizadores.

Relativamente às tecnologias XML e XSL utilizadas no desenvolvimento da ferramenta verifica-se que a linguagem XML, tem sido cada vez mais utilizada para armazenamento de dados para *web*. A utilização da informação através de ficheiros no formato XML permite manter a informação estruturada e interpretá-la facilmente. Este formato permite que sistemas distintos comuniquem entre si de uma forma segura, confiável e fácil. Com este formato é possível separar o conteúdo da forma de apresentação e, desta forma, apresentar o conteúdo de diversas formas XSL. Desta forma as organizações podem alterar a formatação dos seus documentos sem perder a informação e apresenta-la no novo formato.

5.1 Perspectivas de Desenvolvimento

De entre os vários desenvolvimentos que se podem desenvolver na sequência deste trabalho são referidos os seguintes.

Uma primeira linha de estudo poderá ser o desenvolvimento de uma solução que permite desenvolver relatórios dos sistemas ERP sem que os utilizadores sintam a necessidade de conhecimento sobre SQL.

Os relatórios produzidos pela ferramenta são apresentados no formato de tabelas, desta forma poderá ser desenvolvido uma funcionalidade que possibilite a representação e análise gráfica dos relatórios produzidos.

Pode-se introduzir também o conceito de template na configuração dos modelos de documentos. Existência de estruturas de documentos pré-definidas permitindo ao utilizador seleccionar uma estrutura desejada.

Bibliografia

- [1] 1keydata. (2008, May). Reporting Tool Selection. 1keydata.com. Available: <http://www.1keydata.com/datawarehousing/toolreporting.html>
- [2] Wikipédia. (2008). OLAP, Framework. Wikipédia a enciclopédia livre. Available: <http://pt.wikipedia.org>
- [3] W. Du and A. Elmagarmid, "Workow Management: State of the Art vs. State of the Products", HP Labs Technical Report, July, 1997.
- [4] A. Afonso, *A Linguagem XML e Especificações Associadas*, Portugal: FCA Editora, 2003.
- [5] J. Atlee, "Workflow Management Systems - A Standard Architecture for Automating Workflow", University of Waterloo, 1997.
- [6] R. M. Barros, "XML - XSL-FO", CIn-UFPE, 2004.
- [7] Chaumier, *J. Systemes d'information: marché et technologies*, France: Enterprise Moderne, 1986.
- [8] I. Chiavenato, *Gestão de Pessoas: O novo papel dos recursos humanos nas organizações*, Brasil: Campus, 1999.
- [9] J. Parreira e M. Teixeira, "Porquê Gestão Documental?", Dalera Ciberguia, July, 2006.
- [10] World Wide Web Consortium, "XML Pointer Language", World Wide Web Consortium Working Draft, Mar. 1998.

- [11] Armazém de Dados. (2007). Gestão de Documentos. Available: <http://www.armazemdedados.com.br/x.html>
- [12] Dojo. (2008). Dojo, the javascript toolkit. The Dojo Foundation. Available: <http://dojotoolkit.org>
- [13] B. DuCharme, *XSLT: Guia Prático*, Rio de Janeiro, Brasil: Ciência Moderna, 2002.
- [14] F. Silva e J. A. Alves, *ERP e CRM*, 1ª ed. Lisboa, Portugal: Edições Centro Atlântico, 2000.
- [15] M. Francesconi, "Padrões XML para Gestão de Processos de Negócio", USP - FEA - FIA, Março. 2002.
- [16] K. Y. Fung, *XSLT - Interagindo com XML e HTML*, 1ª ed., Ciência Moderna, 2001.
- [17] H. Lie and B.t Bos. (1999, Jan.). Cascading Style Sheets, level 1. W3C. Available: <http://www.w3.org/TR/REC-CSS1>.
- [18] T. Turner and R. Hall, "Eletronic Workflow Using the World Wide Web", University of Florida, 2007.
- [19] E. R. Harold, *XML Bible*, IDG Books, 1999.
- [20] J. Ramalho e P. Henriques, *XML e XSL da Teoria à Prática*, Portugal: FCA Editora, 2002.
- [21] D. Hollingsworth, "The Workflow Management Coalition Specification", Workflow Management Coalition, Janeiro. 1995.
- [22] *IPBrick - Manual de Referência*, ed. 4.3, iPortalMais - Serviços de Internet e Redes, Porto, 2008.
- [23] iPortalMais. (2008). IPBrick. iPortalMais - Serviços de Internet e Redes. Available: <http://www.ipbrick.com>
- [24] *Manual do iPortaDoc*, iPortalMais - Serviços de Internet e Redes, Porto, 2007.

- [25] A. Joaquim. (2005, Fev. 18). Gestão Documental Ganha Maturidade. *Semana Informática*.
- [26] J.Resig and the jQuery team. (2008). jQuery is a new type of JavaScript library. jQuery. Available: <http://jquery.com>
- [27] M. Kay, *XSLT: Referência do Programador*, 2^a ed. Rio de Janeiro, Brasil: Alta Books, 2002.
- [28] P. Lawrence, *Workflow Handbook*, WPMC, 1997.
- [29] T. C. Padilhae e F. S. Martins. (2005, Jan./Apr.). ERP Systems: Characteristics, Implementation Cost, Tendencies. *Revista Produção*.
- [30] C. Milfont. (2007). Frameworks Ajax. CMilfont Tech. Available: <http://www.milfont.org/tech/2007/10/11/frameworks-ajax>
- [31] C. Mohan, "Products, Standards and Research", Proceedings of the NATO Advanced Study Institute on Workflow Management Systems and Interoperability, 1997.
- [32] M. M. Moro. (1998, Set.). Wokflow na Web. UFRGS. Available: <http://www.inf.ufrgs.br/mirella/workflow>
- [33] P. Andrew, J. Conard J. Flanders and S. Woodgate, beta ed., *Windows Workflow Foundation*, SAMS, 2005.
- [34] D. Pawson, *XSL-FO*, 1^a ed. Gravenstein Highway North: O'REILLY, 2002.
- [35] L.G. Silva e M.S. Pessoa, "Gestão da Informação - Uma Visão dos Sistemas ERP", VI SIMPEP - Simpósio de Engenharia de Produção, 1999.
- [36] A. Pires, *eXtensible Styles Formatting Object Seminário 2004.1*, O'REILLY, 2002.
- [37] Prototype. (2007). Prototype JavaScript Framework. Prototype Core Team. Available: <http://www.prototypejs.org>

- [38] J. Resig. (2007, Oct.). JavaScript Libraries. ejohn.org. Available: <http://ejohn.org/blog/javascript-library-overview>
- [39] A. Ricks, "La administracion de documentos como función archivística", in *Congr. Internacional de Artigos*, Washington, 1976.
- [40] Tefko Saracevic. Information Science. (1999, Apr.). *Journal of the American Society for Information Science*.
- [41] M. M. Pedro e A. Sezinando. (2004, nº28). Sistemas de Gestão de Documentos Electrónicos. *Informação & Informática*.
- [42] C. Sousa. (2004). ERP. ISMAG/ISHT. Available: <http://trabalhoerpismag2004.no.sapo.pt>
- [43] W3Schools. (2007). XSL-FO Tutorial. W3S. Available: <http://www.w3schools.com/xslfo>
- [44] W3Schools. (2008). DOM, AJAX. W3Schools. Available: <http://www.w3schools.com>
- [45] The Workflow Management Coalition, "WFMC the Workflow Management Coalition Specification: Workflow Management Coalition Workflow Client, Application (Interface 2) Application Programming Interface (WAPI) Specification", Workflow Management Coalition Beta document, 1997.
- [46] Yahoo. (2008). The Yahoo! User Interface Library. Yahoo!. Available: <http://developer.yahoo.com/yui>