

Faculdade de Engenharia da Universidade do Porto



Ambiente Colaborativo para Apoio à Gestão de Projectos

Helder Filipe Lourenço Marques

Tese submetida no âmbito do
Mestrado Integrado em Engenharia Electrotécnica e de Computadores
Major de Automação

Orientador: Prof. Dr. José António Rodrigues Pereira de Faria

A Dissertação intitulada

“Ambiente colaborativo para apoio à gestão de projectos”

foi aprovada em provas realizadas em 17/Julho/2008

o júri

Presidente

Professor Doutor Américo Lopes de Azevedo
Professor Associado da Faculdade de Engenharia da Universidade do Porto



Professor Doutor Ricardo Jorge Silvério Magalhães Machado
Professor Auxiliar da Escola de Engenharia da Universidade do Minho

✓ 

Professor Doutor José António Rodrigues Pereira de Faria
Professor Auxiliar da Faculdade de Engenharia da Universidade do Porto



O autor declara que a presente dissertação (ou relatório de projecto) é da sua exclusiva autoria e foi escrita sem qualquer apoio externo não explicitamente autorizado. Os resultados, ideias, parágrafos, ou outros extractos tomados de ou inspirados em trabalhos de outros autores, e demais referências bibliográficas usadas, são correctamente citados.

Autor - Helder Filipe Lourenço Marques



Faculdade de Engenharia da Universidade do Porto

Resumo

Esta tese teve por objectivo analisar os requisitos e desenvolver um sistema ágil e flexível para apoio à gestão integrada de múltiplos projectos com uma forte componente colaborativa. Pretendia-se um sistema capaz de se adaptar a diferentes tipos de projectos e a diferentes métodos de trabalho das pessoas, permitindo uma configuração das interfaces e das funcionalidades quer a nível individual quer global. O sistema deveria oferecer, a par de interfaces que permitissem a gestão dos projectos individuais, outras interfaces que proporcionassem uma visão agregadora de todos os projectos em curso e que, em particular, permitisse acompanhar de forma expedita os últimos desenvolvimentos de cada um dos projectos.

Na concepção da solução estiveram presentes conceitos provenientes de um conjunto de novos paradigmas relativos a métodos de trabalho de Human Interaction Management, Getting Things Done e State-Flow, de modo a torná-la ágil, flexível e a aproximá-la à forma como as pessoas realmente trabalham.

O sistema desenvolvido engloba as seguintes grandes áreas funcionais: gestão documental, gestão colaborativa, gestão da comunicação, gestão da informação, gestão de utilizadores e gestão do trabalho e é baseado na plataforma Microsoft Office SharePoint Server 2007.

Na fase inicial do projecto foram exploradas, testadas e analisadas de forma aprofundada as características da plataforma e das ferramentas de desenvolvimento associadas. De seguida, foi efectuada uma análise de requisitos para o sistema. Com base nos resultados dessa, foi elaborado o modelo do domínio, identificando os vários tipos de conteúdos e de operações a gerir pelo sistema. Foram testadas várias abordagens para a implementação do sistema, umas baseadas na utilização quase exclusiva das funcionalidades nativas do SharePoint e outras recorrendo ao desenvolvimento de código específico. Para isso foram utilizadas várias ferramentas de desenvolvimento, entre as quais, o Microsoft Office SharePoint Designer 2007 e o Visual Studio 2008.

Abstract

This thesis aimed to analyse the requirements and develop an agile and flexible system to support the integrated management of multiple projects with a strong collaborative component. The aim was to a system capable of adapting to different types of projects and different working methods of these people, allowing a configuration of the interfaces and features either individually or globally. The system should provide, along with interfaces that allow the management of individual projects, other interfaces that offers an aggregative vision of all ongoing projects and, in particular, that allows to keep up with the latest developments of each project.

In the design of the solution were present concepts from a set of new paradigms for the working methods of Human Interaction Management, Getting Things Done and State-Flow, to make it agile, flexible and bringing it to the way of how people really work.

The system developed includes the following major functional areas: document management, collaborative management, communication management, information management, management of users and managing the work and is based on Microsoft Office SharePoint Server 2007.

In the initial phase of the project were explored, tested and analyzed in detail the characteristics of the platform and tools of development partners. Next, was made an analysis of requirements for the system. Based on the results of that, the model was developed, identifying the various types of content and operations to manage the system. We tested several approaches to the implementation of the system, one based almost exclusive use of native features of SharePoint and other using the development of specific code. For that were used various tools for development, including the Microsoft Office SharePoint Designer 2007 and Visual Studio 2008.

Agradecimentos

Em primeiro lugar, quero agradecer aos meus pais por sempre acreditarem em mim e por todo o apoio que me deram.

Um agradecimento especial à minha namorada Marina Sampaio pelo apoio, motivação e ajuda que sempre me deu.

Quero agradecer ao meu amigo Miguel Silva que trabalhou comigo neste projecto e me ajudou na realização do mesmo.

Quero também agradecer ao meu orientador, Prof. José Faria, por toda a ajuda prestada durante a realização deste projecto.

Agradeço aos pais da minha namorada, Albino Sampaio e Filomena Oliveira, pelo apoio dado e por toda a preocupação que tiveram comigo ao longo desta minha caminhada.

Índice

RESUMO.....	V
ABSTRACT	VII
AGRADECIMENTOS	IX
ÍNDICE	XI
LISTA DE FIGURAS	XV
LISTA DE TABELAS	XVII
ABREVIATURAS	XIX
CAPÍTULO 1	1
INTRODUÇÃO.....	1
1.1. CONTEXTO.....	1
1.2. OBJECTIVOS	1
1.3. METODOLOGIA	2
1.4. ORGANIZAÇÃO DO DOCUMENTO	3
CAPÍTULO 2	5
FUNDAMENTOS DE GESTÃO DE PROJECTOS.....	5
2.1. DEFINIÇÕES BÁSICAS	5
2.2. SOLUÇÕES ACTUAIS	8
2.2.1. <i>Microsoft Office Project 2007</i>	10
2.2.2. <i>@Task</i>	11
2.2.3. <i>Activity Explorer</i>	12
2.2.4. <i>Humanedj</i>	14
2.3. SÍNTESE	16
CAPÍTULO 3	17
NOVOS PARADIGMAS	17
3.1. CONTEXTO.....	17
3.2. BUSINESS PROCESS MANAGEMENT	18
3.3. HUMAN INTERACTION MANAGEMENT	19
3.4. GETTING THINGS DONE	22
3.4.1. <i>Descrição geral do GTD</i>	23
3.4.2. <i>Extensão do GTD para apoiar o trabalho colaborativo</i>	26
3.5. STATE-FLOW	27
3.6. ACTIVITY CENTRIC.....	28
3.7. FUNDAMENTOS COGNITIVOS DO CONHECIMENTO DO TRABALHO	29

3.8. ASPECTOS HUMANOS	30
3.9. SÍNTESE	32
CAPÍTULO 4	35
ANÁLISE FUNCIONAL E ESPECIFICAÇÃO	35
4.1. CONTEXTO	35
4.2. ANÁLISE DE REQUISITOS	36
4.3. MODELO CONCEPTUAL DO DOMÍNIO	37
4.4. CASOS DE USO	39
4.4.1. <i>Gestão Documental</i>	41
4.4.2. <i>Gestão Colaborativa</i>	42
4.4.3. <i>Gestão da Comunicação</i>	43
4.4.4. <i>Gestão da Informação</i>	45
4.4.5. <i>Gestão de Utilizadores</i>	47
4.4.6. <i>Gestão do Projecto</i>	48
4.4.7. <i>Gestão da Personalização</i>	50
4.5. SÍNTESE	51
CAPÍTULO 5	53
ESTUDO E ANÁLISE DO SHAREPOINT	53
5.1. VISÃO GERAL	53
5.2. ARQUITECTURA	55
5.3. ESTRUTURA DE UM PORTAL EM SHAREPOINT	56
5.4. CONTEÚDOS BÁSICOS DO SHAREPOINT	57
5.4.1. <i>Tipos de Sites</i>	57
5.4.2. <i>Tipos de Conteúdo</i>	58
5.4.3. <i>Listas</i>	59
5.4.4. <i>Bibliotecas</i>	60
5.4.5. <i>Vistas</i>	62
5.4.6. <i>Páginas</i>	65
5.4.7. <i>Peças Web</i>	65
5.4.8. <i>Audiências</i>	69
5.4.9. <i>Fluxos de Trabalho</i>	69
5.4.10. <i>Reciclagem</i>	70
5.4.11. <i>Permissões</i>	71
5.4.12. <i>Hierarquia e herança</i>	72
5.4.13. <i>Pesquisa</i>	73
5.4.14. <i>Integração com o Microsoft Office 2007</i>	74
5.4.15. <i>Business Intelligence</i>	76
5.5. FUNÇÕES AVANÇADAS DO SHAREPOINT	76
5.6. SÍNTESE	77
CAPÍTULO 6	79
DESENHO E IMPLEMENTAÇÃO	79
6.1. ABORDAGENS	79
6.2. DESENHO	82
6.3. IMPLEMENTAÇÃO	90
6.4. SÍNTESE	90
CAPÍTULO 7	93
CONCLUSÕES	93
7.1. PRINCIPAIS CONCLUSÕES	93
7.2. PERSPECTIVAS DE DESENVOLVIMENTO	97
REFERÊNCIAS	99

ANEXOS.....	103
A. SOLUÇÕES PERSONALIZADAS PARA SHAREPOINT	105
<i>Centro de Administração</i>	105
<i>O Comando STSADM</i>	109
<i>Cópia de Segurança e Restauro</i>	109
<i>Gestão de Peças Web</i>	110
<i>Personalização</i>	110
<i>Estrutura de um Portal em SharePoint</i>	111
<i>Página Mestra e Esquema de Página</i>	112
<i>Microsoft Office SharePoint Designer 2007</i>	112
<i>Estrutura de Pastas do SharePoint Designer</i>	113
<i>Personalização de uma Página Mestra</i>	115
<i>Formulários</i>	116
<i>Utilização de Custom List Form</i>	116
<i>XSLT</i>	118
<i>Fluxos de Trabalho</i>	119
<i>Ligações de Peças Web</i>	120
<i>Gestão de Tipos de Ficheiros Bloqueados pelo SharePoint</i>	121
<i>Personalização de Ícones no SharePoint</i>	121
<i>Visual Studio</i>	122
<i>Peças Web Personalizadas Desenvolvidas em Visual Studio</i>	123
<i>Exemplo de uma Peça Web que Acede a uma Lista</i>	127
<i>Exemplo de uma Peça Web com AJAX</i>	129
<i>Exemplo de uma Peça Web com Ligações</i>	131
<i>Event Handlers</i>	137
<i>AJAX no SharePoint</i>	138
<i>CAML</i>	141
<i>LINQ para SharePoint</i>	142
<i>SmartPart</i>	142

Lista de figuras

FIGURA 1 – INTERFACE DO MICROSOFT OFFICE PROJECT 2007	11
FIGURA.2 – INTERFACE DO @ TASK	12
FIGURA 3 – INTERFACE DO ACTIVITY EXPLORER.....	13
FIGURA 4 – TRIÂNGULO DO HUMANEDJ.....	14
FIGURA 5 – INTERFACES DO HUMANEDJ	15
FIGURA 6 – FLUXOGRAMA DO <i>GETTING THINGS DONE</i>	23
FIGURA 7 – DIAGRAMA DA SUBJECTIVIDADE DA SEMÂNTICA	31
FIGURA 8 – INTERFACE DO PROBIS.....	31
FIGURA 9 – MODELO CONCEPTUAL DO DOMÍNIO.....	38
FIGURA 10 – PACOTE DE CASOS DE USO DA GESTÃO MULTI-PROJECTOS	41
FIGURA 11 – CASOS DE USO DA GESTÃO DOCUMENTAL.....	42
FIGURA 12 – CASOS DE USOS DA GESTÃO COLABORATIVA	43
FIGURA 13 – CASOS DE USO DA GESTÃO DA COLABORAÇÃO	44
FIGURA 14 – CASOS DE USO DA GESTÃO DA INFORMAÇÃO	46
FIGURA 15 – CASOS DE USO DA GESTÃO DE UTILIZADORES	47
FIGURA 16 – CASOS DE USO DA GESTÃO DO PROJECTO.....	49
FIGURA 17 – CASOS DE USO DA GESTÃO DA PERSONALIZAÇÃO	50
FIGURA 18 – ARQUITECTURA DO SHAREPOINT.....	56
FIGURA 19 – ESTRUTURA DE UM PORTAL EM SHAREPOINT	57
FIGURA 20 – VERSÃO <i>MOBILE</i> DE UM SITE SHAREPOINT NUM PDA	58
FIGURA 21 – LISTAS DO SHAREPOINT	60
FIGURA 22 – BIBLIOTECAS DO SHAREPOINT.....	61
FIGURA 23 – ESTILOS DE PERSONALIZAÇÃO DE LISTAS.....	63
FIGURA 24 – VISTAS DE CALENDÁRIO DO SHAREPOINT.....	64
FIGURA 25 – VISTA DE GANTT DO SHAREPOINT	64
FIGURA 26 – PÁGINA DE SHAREPOINT EM MODO EDITÁVEL.....	66
FIGURA 27 – PEÇA WEB IMAGEM.....	67
FIGURA 28 – MENU DE PERSONALIZAÇÃO DE UMA PEÇA WEB	68
FIGURA 29 – TIPOS DE PERMISSÕES DO SHAREPOINT.....	72
FIGURA 30 – HERANÇA DE PERMISSÕES	73
FIGURA 31 – RESULTADOS DA PESQUISA DO SHAREPOINT	74
FIGURA 32 – INTEGRAÇÃO DO SHAREPOINT COM O OFFICE	75
FIGURA 33 – INTERFACE DA SEGUNDA ABORDAGEM AO SHAREPOINT	81
FIGURA 34 – INTERFACE DO MODELO A IMPLEMENTAR	83
FIGURA 35 – EXEMPLO DE NAVEGAÇÃO ENTRE AS ÁREAS DE TRABALHO DO MODELO	84
FIGURA 36 – LISTA DE RELAÇÃO DO MODELO.....	86
FIGURA 37 – MODELO DE LISTAS E SUAS RELAÇÕES.....	87
FIGURA 38 – PÁGINA INICIAL DO CENTRO DE ADMINISTRAÇÃO	105
FIGURA 39 – ARQUITECTURA DA <i>WEB APPLICATION</i>	106
FIGURA 40 – PÁGINA DE <i>OPERATIONS</i> DO CENTRO DE ADMINISTRAÇÃO	106

FIGURA 41 – PÁGINA DE <i>APPLICATION MANAGEMENT</i> DO CENTRO DE ADMINISTRAÇÃO	107
FIGURA 42 – PÁGINA DE CONFIGURAÇÃO DA PESQUISA NO CENTRO DE ADMINISTRAÇÃO	108
FIGURA 43 – CENÁRIO DE DESENVOLVIMENTO E PRODUÇÃO	111
FIGURA 44 – MAPEAMENTO DOS OBJECTOS DO SHAREPOINT PARA O SITE	111
FIGURA 45 – RELAÇÃO ENTRE <i>MASTER PAGE</i> E <i>PAGE LAYOUT</i>	112
FIGURA 46 – INTERFACE DO SHAREPOINT DESIGNER	113
FIGURA 47 – RELAÇÃO DA ARQUITECTURA DE PASTAS DO SHAREPOINT DESIGNER COM OS OBJECTOS DO SHAREPOINT	114
FIGURA 48 – QUADRO DE PROPRIEDADES DE UMA LISTA	115
FIGURA 49 – EXEMPLO DE PERSONALIZAÇÃO DE UMA <i>MASTER PAGE</i>	115
FIGURA 50 – EXEMPLO DE FORMULÁRIOS PERSONALIZADOS NO SHAREPOINT.....	116
FIGURA 51 – QUADRO DE INSERÇÃO DE UM NOVO FORMULÁRIO PERSONALIZADO	117
FIGURA 52 – CAMPOS DE INSERÇÃO DE UM NOVO FORMULÁRIO PERSONALIZADO	117
FIGURA 53 – EXEMPLO DE UTILIZAÇÃO DE XSLT.....	118
FIGURA 54 – QUADRO DE DESENVOLVIMENTO DE FUNÇÕES XSLT	119
FIGURA 55 – EXEMPLO DE CRIAÇÃO DE UM FLUXO DE TRABALHO NO SHAREPOINT DESIGNER.....	120
FIGURA 56 – EXEMPLO DE LIGAÇÕES ENTRE PEÇAS WEB	121
FIGURA 57 – EXEMPLO DE PERSONALIZAÇÃO DE ÍCONES NO SHAREPOINT	122
FIGURA 58 – FICHEIROS NECESSÁRIOS PARA A CRIAÇÃO DE UMA PEÇA WEB EM VISUAL STUDIO	125
FIGURA 59 – EXEMPLO DE FUNCIONAMENTO DE UMA PEÇA WEB DESENVOLVIDA EM VISUAL STUDIO	127
FIGURA 60 – EXEMPLO DE UMA PEÇA WEB PERSONALIZADA QUE ACEDE A UMA LISTA.....	128
FIGURA 61 – EXEMPLO DE UMA PEÇA WEB QUE UTILIZA AJAX	130
FIGURA 62 – EXEMPLO DE UMA PEÇA WEB <i>PROVIDER</i> PARA A LIGAÇÃO DE PEÇAS WEB.....	131
FIGURA 63 – EXEMPLO DE CONFIGURAÇÃO DA LIGAÇÃO ENTRE PEÇAS WEB.....	133
FIGURA 64 – EXEMPLO DE PEÇAS WEB PERSONALIZADAS QUE IMPLEMENTAM CONEXÕES	134
FIGURA 65 – LISTA FONTE DE DADOS PARA A PEÇA WEB <i>PROVIDER</i> PERSONALIZADA E COM AJAX.....	135
FIGURA 66 – EXEMPLO DO FUNCIONAMENTO DE PEÇAS WEB LIGADAS E ACTUALIZADAS COM AJAX..	136

Lista de tabelas

TABELA 1 – COMPARAÇÃO ENTRE AS FERRAMENTAS HUMANEDJ E OUTLOOK	15
TABELA 2 – TABELA DE ENTIDADES, OBJECTOS E CORRESPONDÊNCIA COM OS OBJECTOS DE HIM.....	38
TABELA 3 – MAPEAMENTO DE ÁREAS DE TRABALHO PARA OBJECTOS DE SHAREPOINT	82
TABELA 4 – MAPEAMENTO DE MÓDULOS PARA OBJECTOS DE SHAREPOINT.....	83
TABELA 5 – TIPO DE LISTA E COLUNAS IMPLEMENTADAS NA LISTA DE PROJECTOS	87
TABELA 6 – TIPO DE LISTA E COLUNAS IMPLEMENTADAS NA LISTA DE FASES.....	87
TABELA 7 – TIPO DE LISTA E COLUNAS IMPLEMENTADAS NA LISTA DE EVENTOS.....	88
TABELA 8 – TIPO DE LISTA E COLUNAS IMPLEMENTADAS NA BIBLIOTECA DE DOCUMENTOS DOCUMENTOS.....	88
TABELA 9 – TIPO DE LISTA E COLUNAS IMPLEMENTADAS NA LISTA DE ANOTAÇÕES.....	88
TABELA 10 – TIPO DE LISTA E COLUNAS IMPLEMENTADAS NA LISTA DE E-MAILS	88
TABELA 11 – TIPO DE LISTA E COLUNAS IMPLEMENTADAS NA LISTA DE CONTACTOS	89
TABELA 12 – TIPO DE LISTA E COLUNAS IMPLEMENTADAS NA LISTA DE PROBLEMAS.....	89
TABELA 13 – TIPO DE LISTA E COLUNAS IMPLEMENTADAS NA LISTA DE RELAÇÃO.....	89
TABELA 14 – TIPO DE LISTA E COLUNAS IMPLEMENTADAS NA LISTA DE MENSAGENS	89

Abreviaturas

AD	Active Directory
ASP	Active Server Pages
AE	Activity Explorer
API	Application Programming Interface
AJAX	Asynchronous JavaScript and XML
BI	Business Intelligence
BPEL	Business Process Execution Language
BPM	Business Process Management
BPMS	Business Process Management System
BPMN	Business Process Modeling Notation
C#	C Sharp
CSS	Cascading Style Sheets
CAML	Collaborative Application Markup Language
CT	Content Types
DNS	Domain Name System
XML	Extensible Markup Language
XSLT	Extensible StyleSheet Language Transformations
GTD	Getting Things Done
HIM	Human Interaction Management
HIMS	Human Interaction Management System
HTML	HyperText Markup Language
IIS	Internet Information Services
KPI	Key Performance Indicators
LINQ	Language Integrated Query
MOSS	Microsoft Office SharePoint Server
PDA	Personal Digital Assistant
PDF	Portable Document Format
PMI	Project Management Institute
RSS	Really Simple Syndication
STM	Serviços Técnicos de Manutenção
SP	SharePoint
SGQ	Sistema de Gestão da Qualidade
SF	State-Flow

SQL	Structured Query Language
SAP	Systems Applications and Products
TI	Tecnologias da Informação
TQM	Total Quality Management
UML	Unified Modeling Language
URL	Uniform Resource Location
VS	Visual Studio
WYSIWYG	What You See Is What You Get
WSS	Windows SharePoint Services
WWF	Windows Workflow Foundation

Capítulo 1

Introdução

Neste capítulo são apresentados o contexto em que se desenvolveu este projecto, os objectivos do mesmo e a metodologia seguida para este problema.

1.1. Contexto

Este projecto desenvolveu-se no âmbito do Gabinete de Qualidade da FEUP (QualiFEUP). O Gabinete tem por missão apoiar os Serviços Centrais e os Departamentos da FEUP no desenvolvimento de Sistemas de Gestão da Qualidade, SGQ, adoptando para tal uma metodologia baseada na realização de projectos de melhoria por equipas constituídas por colaboradores dos serviços e apoiada por técnicos do QualiFEUP.

1.2. Objectivos

O projecto aqui apresentado teve por objectivo estudar, analisar e desenvolver um sistema flexível para apoiar a gestão dos projectos da responsabilidade dos Serviços Técnicos e de Manutenção da FEUP (STM).

Pretendeu-se desenvolver um sistema capaz de suportar a gestão de projectos com uma forte componente colaborativa. O sistema deveria ser desenvolvido em torno de um espaço de trabalho partilhado, oferecendo o seguinte conjunto de funcionalidades:

- Gestão da documentação
- Gestão do trabalho
- Gestão da comunicação
- Gestão da informação
- Personalização da área de trabalho pessoal

Pretendia-se também que o sistema a desenvolver tivesse capacidade para se adaptar a diferentes padrões de projectos, desde projectos simples e desestruturados até projectos mais complexos e estruturados, permitisse uma configuração em função das características particulares de cada equipa e de cada projecto, apresentasse uma área agregadora onde figurem os últimos desenvolvimentos dos vários projectos em curso e fosse capaz de evoluir em função das necessidades e requisitos de cada equipa.

Fez ainda parte dos objectivos deste projecto adquirir um conhecimento sobre a ferramenta Microsoft Office SharePoint Server 2007 e as ferramentas de desenvolvimento associadas que permitisse avaliar esta plataforma não apenas na perspectiva deste projecto mas também na perspectiva da sua eventual utilização noutros projectos QualiFEUP. Nessa avaliação foram considerados aspectos como as funcionalidades predefinidas que oferece, robustez, fiabilidade, facilidade de utilização, necessidade de desenvolvimento de código externo, etc.

1.3. Metodologia

No desenvolvimento do projecto foram adoptados conceitos e métodos da engenharia de sistemas, na perspectiva de garantir a satisfação das necessidades dos utilizadores. Por conseguinte, este decorreu de acordo com as seguintes fases:

- Análise do estado do problema
- Investigação de alternativas
- Modelo do sistema
- Integração
- Entrada em funcionamento do sistema
- Análise do desempenho
- Re-avaliação
- Documentação das soluções implementadas

Numa primeira fase do projecto, foi feito um estudo sobre gestão de projectos que incidiu sobre diferentes tipos de projectos e de ferramentas existentes no mercado, com o objectivo de identificar padrões na gestão de projectos. Esta fase foi fundamentalmente sustentada por pesquisas e leituras de artigos, livros e outras referências relacionadas com o tema.

Paralelamente foi efectuada uma investigação sobre os conceitos e paradigmas relativos aos métodos de trabalho e de interacção entre as pessoas, no âmbito de processos colaborativos, e que têm por objectivo comum melhorar o desempenho e a produtividade. Como se verá, esses paradigmas estão na base da solução que veio a ser adoptada.

Seguidamente, foi efectuado um levantamento e análise de requisitos para o sistema a desenvolver. Durante esta fase foram efectuadas reuniões com os técnicos dos STM, futuros utilizadores do sistema, para ser possível perceber os métodos de trabalho actuais, as suas necessidades e as funcionalidades que gostariam de ver incluídas num sistema de gestão de projectos. Durante esta fase, procurou-se recolher o máximo de informação e material possíveis, sem ambiguidades, para se estar ciente do problema em concreto.

De seguida, foi efectuada uma análise de requisitos para o sistema. Com base nos resultados dessa análise, foi elaborado o modelo do domínio e foram definidos os principais conteúdos e as grandes funcionalidades do sistema.

Foi estudada e analisada a ferramenta Microsoft Office SharePoint Server 2007, MOSS, como plataforma de suporte para este projecto e eventuais projectos futuros. Isto incluiu a análise da necessidade de utilização das ferramentas de desenvolvimento associadas como o SharePoint Designer 2007 para personalização e o Visual Studio 2008 para o desenvolvimento de código externo. Nesta perspectiva, foram estudadas três alternativas diferentes, uma com recurso ao desenvolvimento de *User Controls* e a utilização de uma base de dados externa, outra baseada na utilização quase exclusiva das funcionalidades nativas do SharePoint, SP, e a última recorrendo ao desenvolvimento de código específico com acesso aos objectos do SharePoint.

Por fim, foi desenvolvido um modelo, que respondesse aos requisitos especificados, influenciado pelos resultados da análise do SharePoint.

1.4. Organização do Documento

Este documento está organizado em sete capítulos, o primeiro dos quais esta introdução.

O segundo capítulo expõe as definições de Projecto e de Gestão de Projectos, apresentando as várias características de um projecto, as fases típicas que o constituem, bem como as funções de um gestor de projectos e da sua equipa de trabalho. São apresentadas também as soluções actuais de softwares de gestão de projectos existentes no mercado, com uma breve explicação sobre as lacunas deste tipo de sistemas informáticos. Por fim, são apresentadas quatro ferramentas em particular, com uma descrição das suas principais funcionalidades.

O capítulo três introduz um conjunto de conceitos e novos paradigmas sobre as formas de trabalho nas organizações e os modos de interacção entre as pessoas. Apresenta ainda o conceito de fluxo de estados de um processo e uma visão sobre o conceito de explorador de actividade.

No quarto capítulo são abordados e analisados os requisitos do sistema que se pretende implementar, definidas as entidades para o modelo conceptual do domínio e especificados os casos de uso que permitem implementar esse modelo.

O capítulo cinco apresenta uma visão geral sobre o SharePoint e uma descrição sumária da estrutura e conteúdos básicos de um portal em SharePoint. Desenvolvimentos mais detalhados sobre soluções mais práticas encontram-se no Anexo A.

O sexto capítulo descreve as três abordagens que foram testadas e avaliadas para a implementação do sistema de gestão de projectos no SP, bem como o modelo desenhado para implementar o sistema e resume o que foi desenvolvido para essa implementação.

O sétimo capítulo apresenta uma conclusão de todo o trabalho efectuado ao longo deste projecto e sugere algumas perspectivas de desenvolvimento para o futuro.

Capítulo 2

Fundamentos de Gestão de Projectos

Este capítulo expõe as definições de Projecto e de Gestão de Projectos, apresentando as várias características de um projecto, as fases típicas que o constituem, bem como as funções de um gestor de projectos e da sua equipa de trabalho. São apresentadas também as soluções actuais de softwares de gestão de projectos existentes no mercado, com uma breve explicação sobre as lacunas deste tipo de sistemas informáticos. Por fim, são apresentadas quatro ferramentas em particular com uma descrição das suas principais funcionalidades.

2.1. Definições Básicas

Segundo o Project Management Institute, PMI, um projecto é um esforço empreendido temporariamente para criar um produto ou serviço único. Isto significa que um projecto tem um início e um fim claramente definidos e que o produto ou serviço criados são de alguma forma diferentes de projecto para projecto, pelo que, não há dois projectos iguais. [1]

Normalmente, um projecto resulta de um relatório ou estudo e, muitas vezes, não passa de uma página escrita na agenda como resultado de uma reunião. Um projecto tem como principais características ser um instrumento de mudança único com um princípio e um fim claramente identificáveis, que produz resultados de acordo com determinados objectivos previamente especificados. Um projecto envolve custos, recursos e tempo, cuja gestão é da responsabilidade de uma pessoa ou de um grupo de pessoas. Estas características não são claras na sua totalidade no início do projecto mas vão sendo definidas e ajustadas ao longo deste. No entanto, todos estes pontos devem ser minimamente averiguados no início de cada projecto. [2]

Tipicamente, um projecto atravessa várias fases, entre as quais se destacam:

- Início
- Especificação
- Design
- Implementação
- Instalação
- Operação e revisão

O mesmo instituto, PMI, entende que a gestão de projectos é a aplicação de conhecimento, capacidades, ferramentas e técnicas às actividades do projecto, que vão de encontro às necessidades do mesmo. A gestão de projectos abrange várias áreas como a gestão do tempo, custos, qualidade, recursos humanos, comunicações, integração, ambiente, risco e intervenção. [1]

Um gestor de projectos tem, essencialmente, as seguintes funções:

- Planear
- Organizar
- Coordenar
- Controlar
- Liderar

Por conseguinte, o gabinete de projectos, com os seus vários intervenientes, deve preparar planos de actividades a serem executadas e manter informados os demais intervenientes do projecto sobre a evolução deste, em relação aos planos definidos, através de relatórios de evolução. Este gabinete deve ainda ser capaz de localizar e resolver possíveis problemas, estando preparado para eventuais evoluções que sejam necessárias alterar ou acrescentar ao plano inicial do projecto. É da responsabilidade do gabinete de projectos o controlo de prazos, o arquivo de toda a documentação, incluindo a preparação e elaboração de normas e controlos de qualidade, o registo da evolução do projecto e a organização de reuniões bem como a redacção das respectivas actas. O controlo na gestão de projectos deve ser caracterizado como o processo que, na realidade, se destina a fornecer um mecanismo de identificação dos problemas, através do qual se podem procurar e implementar soluções. [2]

Num projecto, as tarefas devem resultar num produto final ou serviço, ter um responsável, datas de início e conclusão claramente definidas e ser mensuráveis em termos de custos, recursos e tempo.

O registo de dados básicos acerca do progresso das actividades permite obter uma grande quantidade de informação, auxiliadora na interpretação da evolução dessas mesmas actividades. Há vários níveis de controlo na gestão de projectos, entre os quais se destacam o

controlo do tempo, custos e qualidade, que compõem o triângulo do *Total Quality Management*, TQM.

No controlo do tempo, apesar de ser importante ter presente a situação actual do projecto, isto é, a quantidade de tempo dispendida até ao momento e a sua relação com a progressão planeada, a informação mais valiosa é saber o tempo disponível até à conclusão planeada do projecto e a quantidade de actividades ainda por realizar. Só assim se pode ter uma noção real do estado do projecto, ou seja, se este se encontra atrasado ou não. Em termos estatísticos, é relevante ter informações, sob a forma de percentagem, sobre o tempo dispendido até ao momento e o respectivo saldo restante, as variações de tempo gasto em relação ao planeado e a estimativa de tempo total que será consumido quando se atingir o fim do projecto.

Em termos de custos, o controlo é semelhante ao anterior com a alteração das variáveis a medir. Neste caso, importa saber o custo actual do projecto bem como o saldo restante, as variações de custos sofridas em relação ao orçamento previsto e ainda estimativas sobre o custo total do projecto.

Ao contrário do tempo e dos custos, para os quais há unidades de medida como dias ou euros, a bitola pela qual se mede a qualidade não é assim tão fácil de estabelecer, razão pela qual é muitas vezes esquecida. O estabelecimento de normas de qualidade envolve a definição clara daquilo que o produto final ou serviço deve ser no fim de cada actividade, incluindo todos os factores de qualidade que possam ser apropriados. As definições de qualidade são tão variadas quanto os projectos, mas compreendem considerações como segurança, durabilidade, precisão e funcionalidade. Se os factores de qualidade são definíveis, então também devem ser mensuráveis de alguma forma, seja pela frequência com que fracassam, pelo tempo de vida ou pelo número de correcções que exijam.

A avaliação do tempo decorrido, dos custos e da qualidade requer que o gestor de projectos tenha um conhecimento detalhado da evolução de todas as actividades que estão a decorrer. Existem diversas formas de recolher essa informação, seja através de relatórios de evolução periódicos, reuniões de evolução ou até mesmo de forma directa, andando pelo espaço do projecto, situação esta menos habitual dada a agenda típica de um gestor de projectos.

Os relatórios de evolução devem ser resumidos e elaborados, idealmente uma vez por semana, por todos aqueles que têm à sua responsabilidade a execução de uma actividade. Estes devem revelar o trabalho desenvolvido durante um determinado período de tempo, os desvios ao plano, o trabalho a fazer no período seguinte e os problemas de que se tenha conhecimento. Caso haja algum problema, este deve ser logo reportado com a descrição do mesmo, as actividades directamente afectadas, os impactos em termos de tempo, custos e qualidade, uma acção proposta, caso exista, e eventuais comentários.

Na finalização do projecto, é aconselhável tomar algumas notas para futura consulta rápida. Estas podem avaliar o sucesso do projecto, determinar a necessidade de trabalho

complementar e identificar as lições aprendidas, em jeito de conclusão, para algumas situações menos desejáveis não se repetirem novamente.

2.2. Soluções Actuais

O desenvolvimento de sistemas informáticos de gestão de projectos transformou, ao longo dos anos, a forma de trabalhar dos gestores de projectos. Graças à capacidade que estas aplicações têm para tratar grandes quantidades de informação, por vezes bastante complexa, conseguem tirar de cima do gestor muitas horas de trabalho enfadonho. À medida que a tecnologia evolui, mais sofisticados se tornam estes programas, uma vez que conseguem não só armazenar e manipular grandes quantidades de informação, como também permitem fazer rápidas alterações aos planos.

O planeamento é um processo repetitivo no qual se fazem mudanças contínuas para se obter inicialmente o plano óptimo e, posteriormente, para se adaptar a novas realidades. Ao utilizar uma aplicação informática, é possível fazer rápidos testes no campo hipotético com a variante condicional "e se" para analisar, por exemplo, qual é o impacto sofrido no projecto com a utilização de mais meios em determinado ponto do plano.

Através de relatórios electrónicos é possível também uma partilha rápida e eficaz de informação detalhada relativamente a desenvolvimentos do projecto. Os programas actuais oferecem inclusivamente a apresentação de informação, previamente tratada, com uma grande qualidade gráfica. No entanto, apesar de alguns programas serem capazes de efectuar processamentos de dados altamente sofisticados, deixam depois muito a desejar no que respeita à apresentação de conclusões ou relatórios.

A oferta de algumas funcionalidades, por parte de alguns sistemas de gestão de projectos actuais, pode ser perigosa e absorver demasiada atenção e tempo do gestor. Este não deve cair na tentação de dedicar demasiado tempo a garantir que o plano, por vezes com uma grande complexidade, seja perfeito, desviando-se desta forma dos verdadeiros problemas do projecto. Um plano é apenas uma declaração de intenções e passá-lo à prática é realmente a função nobre para o gestor de projectos. Dedicar bastante tempo, no início, ao planeamento do projecto é necessário porque é uma fase muito importante nas funções do gestor mas, todavia, a maior parte do seu tempo é passado a controlar a evolução do trabalho.

De referir que, a utilização de sistemas informáticos sofisticados não pode, no entanto, substituir uma sólida gestão de projectos e, por si só, não garante o sucesso de nenhum projecto.

Até aos dias de hoje, os grandes ganhos de produtividade nalguns processos têm sido atingidos através da formalização dos processos em fluxos de trabalho computadorizados. Contudo, em muitos desses processos não se obteve o resultado esperado com esta abordagem e, em vez disso, os utilizadores têm dependido de ferramentas colaborativas ponto a ponto (*ad hoc*), como por exemplo o e-mail, para coordenarem o seu trabalho.

As mais simples ferramentas de comunicação ponto a ponto, como por exemplo o telefone, o e-mail ou as mensagens instantâneas, são críticas para os trabalhadores que dependem delas em processos não estruturados. De facto, não há forma mais fácil e mais rápida de se obter uma resposta por parte de outros trabalhadores da organização. No entanto, quando se trata de colaboração mais alargada, que envolva várias organizações, estas ferramentas tornam-se rapidamente intratáveis.

Para projectos grandes, foram inventadas ferramentas com áreas de trabalho partilhadas com o intuito de ajudar as equipas a gerirem o conteúdo do projecto de forma partilhada. Para outros casos, tal já não se verifica, pelo que, as pessoas tentam de qualquer forma dar alguma estrutura ao seu trabalho onde, por exemplo, o registo de eventos não passa de simples anotações em blogues ou ferramentas similares. Em suma, assiste-se hoje em dia a uma proliferação de diversas ferramentas que, na sua grande maioria, não se encontram interligadas.

As organizações precisam de re-desenhar e reinstalar os seus processos de forma mais flexível, que reflecta melhor a forma como as pessoas realmente trabalham. O grande desafio, no desenvolvimento de aplicações que suportem estes processos, é não só simplificar e facilitar a integração mas também ter cuidado com o conseqüente problema de difusão, isto é, evitar sobrecarregar o utilizador com excesso de informação, ferramentas e serviços.

No mercado actual existem inúmeros softwares de gestão de projectos, com características muito variadas, e uma pequena lista de softwares mais vocacionados para a componente colaborativa e humana.

No que respeita a aplicações de gestão de projectos, quer as gratuitas quer as licenciadas, podem ser de instalação local ou baseadas na utilização da internet. As de instalação local, normalmente são multi-plataforma e oferecem um ambiente gráfico mais intenso, enquanto que, o outro leque de aplicações, que não implica uma instalação específica a não ser de browser, oferece por norma mais funcionalidades colaborativas. No entanto, a colaboração em modo *offline* e posterior sincronização online não é uma funcionalidade comum nos sistemas actuais.

A integração é uma grande lacuna na grande maioria dos sistemas de gestão de projectos. Por exemplo, a integração com o e-mail, cuja utilização é extremamente frequente e de grande

importância nos dias de hoje, é ignorada por quase todas as ferramentas de gestão de projectos.

A apresentação de informação numa vista de Gantt, que por exemplo permite ter uma visão global do plano do projecto, também não é uma funcionalidade comum neste tipo de ferramentas.

As aplicações que oferecem um vasto leque de funcionalidades, falham na interligação dos seus objectos, não permitindo ao utilizador associar, por exemplo, um documento a uma tarefa.

São também escassas as ferramentas que apresentam uma vista agregada sobre os desenvolvimentos recentes dos vários projectos em curso e, quando apresentam, obrigam o utilizador a entrar dentro do projecto para ver o que de facto aconteceu nos últimos tempos a este.

É difícil encontrar uma aplicação que permita uma fácil personalização da interface do utilizador, com vários parâmetros de configuração.

É ainda mais difícil encontrar uma aplicação que disponibilize uma vista de pastas idêntica ao explorador de ficheiros do Windows que permita uma rápida, cómoda e simples organização dos ficheiros pertencentes ao projecto.

São exemplos de softwares de gestão de projectos o OpenProj, o Kplato, o dotProject, o Project.net, o TeamCenter, o Artemis, o Primavera, o Planisware, o Tracker Suite, o Daptiv, o Instant Business Network, o OpenAir, o Oracle Projects, o Projectplace, o TeamWork, o VPMi e o Zoho Projects, entre muitos outros.

De seguida são apresentadas, com uma breve descrição, as principais funcionalidades de quatro softwares, dois de gestão de projectos e dois com uma forte componente colaborativa e humana, que são, respectivamente, o famoso Microsoft Office Project 2007, o @Task, o Activity Explorer e o Humanedj.

2.2.1. Microsoft Office Project 2007

O Microsoft Office Project 2007 é uma ferramenta de gestão de projectos local muito vocacionada para o planeamento de tarefas e gestão de custos, tempo e recursos. Oferece uma vista de Gantt, que apresenta uma visão geral do plano do projecto, as dependências das tarefas e as restrições de calendário, permitindo testar várias hipóteses de planeamento, de modo a chegar a uma solução óptima. Permite a utilização de vários modelos de relatório visuais para a apresentação gráfica dos respectivos relatórios. O aspecto de um projecto aberto no Project 2007 é o ilustrado na figura 1. [3]

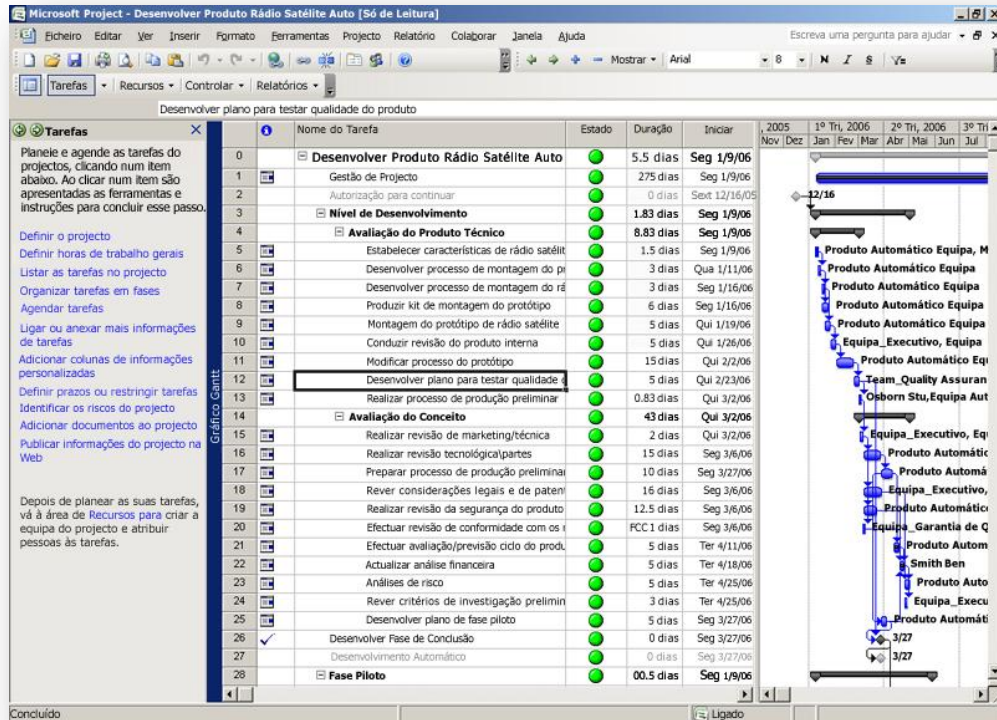


Figura 1 – Interface do Microsoft Office Project 2007

2.2.2. @Task

O @Task é outra ferramenta de gestão de projectos, bastante completa, baseada na Web, que possui como principais funcionalidades a gestão documental, gestão colaborativa, gestão de recursos, tempo e custos e planeamento de tarefas.

Permite a criação de projectos com base em modelos e apresenta uma vista agregada dos vários projectos com a informação sobre o estado actual do mesmo.

Disponibiliza também indicadores estatísticos sobre a situação actual do projecto, uma vista de Gantt para visualizar o plano e possibilita ainda a consulta de projectos e tarefas, oferecendo para tal vários filtros e ordenações.

No entanto, as deficiências apontadas a este tipo de aplicações, no ponto 2.2., continuam a estar presentes nesta ferramenta em particular, ou seja, não é possível associar objectos e a vista agregada de projectos não passa de uma lista com o nome dos mesmos e respectivos estados actuais. A figura 2 ilustra o ambiente de trabalho do @Task. [4]

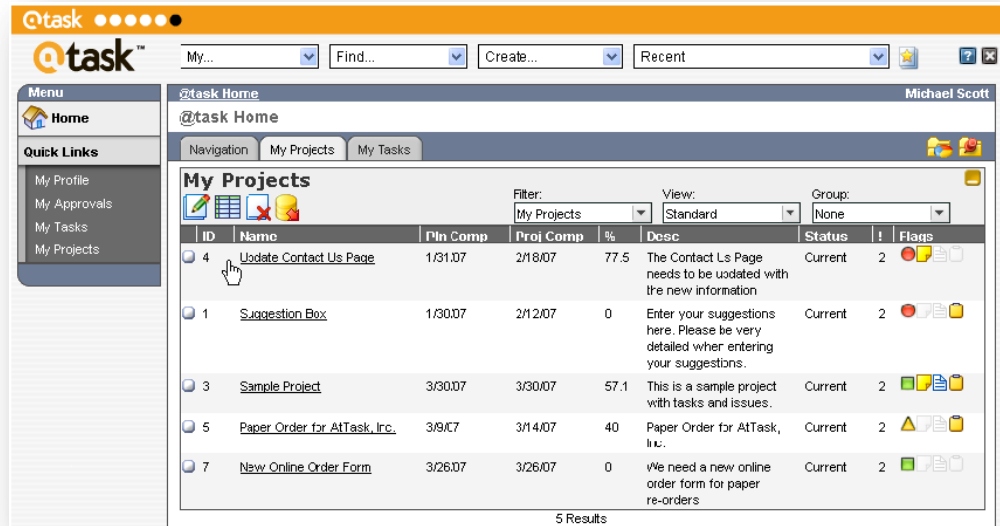


Figura.2 – Interface do @ Task

2.2.3. Activity Explorer

O Activity Explorer, AE, é o primeiro produto da IBM que suporta a noção de colaboração centrada em actividade. Este novo paradigma organiza e integra recursos, ferramentas e pessoas num conceito computacional de actividade de trabalho, com o objectivo de aumentar a qualidade e eficiência do trabalho. Este produto é actualmente executado como uma aplicação separada, com o IBM Workspace Client.

No AE, uma actividade é modelada como um conjunto de objectos relacionados e partilhados que representam uma tarefa ou um projecto. O conjunto de objectos relacionados, denominado por *activity thread*, é estruturado e representa o contexto da tarefa.

Esta ferramenta suporta a partilha de cinco tipos de objectos: mensagens, registos de conversas, ficheiros, pastas e imagens capturadas do ecrã.

Na figura 3 ilustra-se o ambiente de trabalho do Activity Explorer, onde se pode encontrar a lista de actividades (A), que oferece múltiplas vistas bem como a possibilidade de ordenação e filtragem, e a vista em árvore (B), que apresenta todas as actividades do utilizador. O painel de detalhes (C) mostra algumas informações da actividade seleccionada, enquanto que o painel Activity Thread (D) mapeia em árvore os objectos partilhados. O Activity List e o Activity Thread encontram-se sincronizados quando se selecciona um objecto.

Muitas das investigações efectuadas sobre a utilização do e-mail como ferramenta de trabalho tentam encontrar resposta a como adicionar gestão de tarefas ao e-mail. O AE é uma abordagem à gestão das actividades baseada na comunicação, onde as comunicações são colocadas em objectos partilhados em vez de serem directamente entre pessoas mas permitindo, contudo, a troca de e-mails privados num ambiente partilhado. [5]

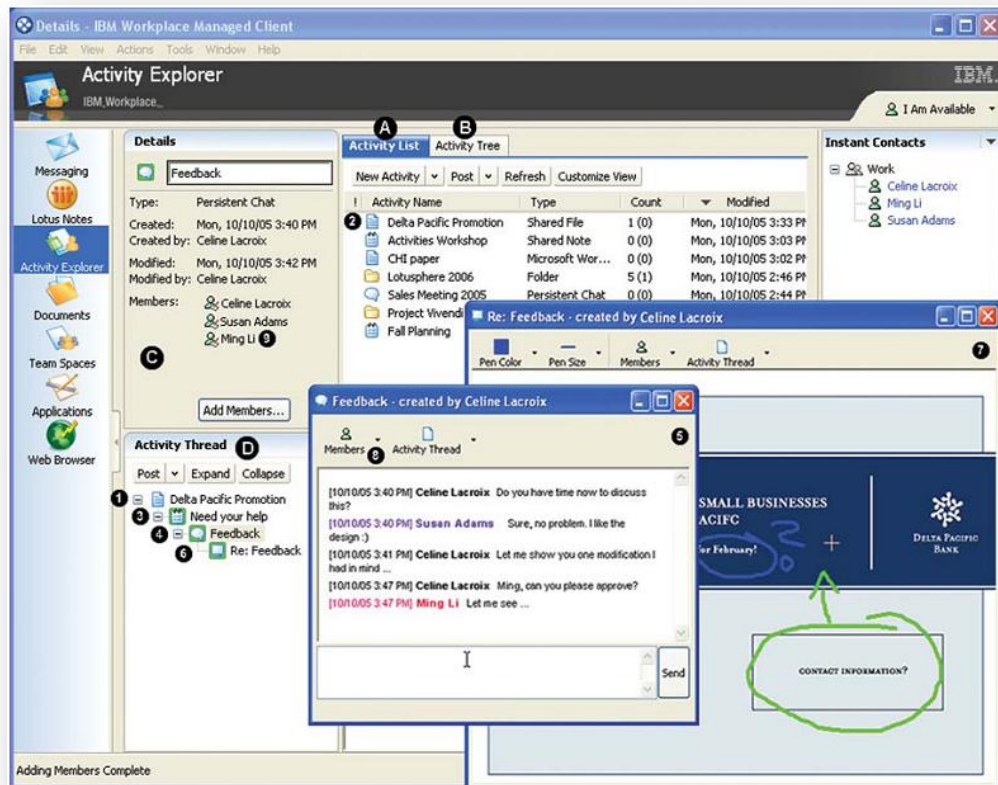


Figura 3 – Interface do Activity Explorer

As pessoas usam o e-mail não só para comunicar mas também para gerir vários tipos de actividades de trabalho. No entanto, estes vários tipos de actividades não são bem suportadas pelo e-mail porque, para actividades que se prolongam por longos períodos de tempo, rapidamente se tornam intratáveis.

A notificação das pessoas sobre as alterações do trabalho é importante para a coordenação da colaboração e, hoje em dia, alguns sistemas disponibilizam algumas áreas de trabalho para facilitar esse trabalho colaborativo. No entanto, estes sistemas carecem de integração com outro tipo de sistemas, como por exemplo de comunicação ou coordenação.

Muitos dos sistemas que proporcionam um robusto suporte para colaboração estruturada são relativamente áduos quando se inicia a sua utilização, o que desencoraja os utilizadores. Na realidade, uma simples actividade colaborativa envolve normalmente várias ferramentas e tecnologias a diferentes níveis de formalidade, fazendo com que as pessoas dispersem a sua atenção.

Uma das dificuldades sentidas pelos utilizadores do Activity Explorer prende-se com o acesso à informação que, segundo os mesmos, é confuso e de difícil pesquisa.

2.2.4. Humanedj

O Humanedj é o primeiro software a ser criado directamente com base nos princípios do Human Interaction Management, HIM, que se encontram descritos no ponto 3.3. Num mundo onde o trabalho está gradualmente a ser automatizado, o trabalho humano tem-se tornado mais importante que nunca. As ferramentas de internet facilitaram a comunicação mas não tornaram a colaboração mais eficiente. O Humanedj é uma ferramenta pessoal com o objectivo de ajudar a colaboração entre pessoas. Esta inclui, por exemplo, suporte para regras de negócio, actos de fala, XML, Web Services, linguagens de *script*, navegação Web e acesso a documentos externos. Para além de ser multi-plataforma e gratuita, é executada na máquina de cada participante do processo e não necessita de qualquer servidor. O Humanedj é visto como um possível substituto dos ambientes de trabalho comuns e, segundo os seus criadores, um utilizador atravessa, normalmente, três fases na adopção do software, tal como se pode ver na figura 4.

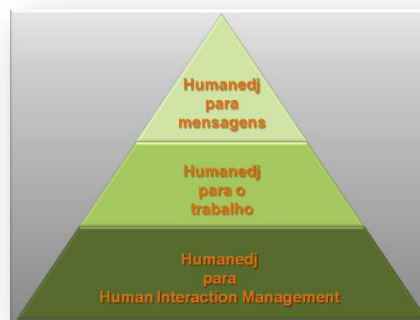


Figura 4 – Triângulo do Humanedj

Na primeira fase, Humanedj para mensagens, a aplicação é usada para simplificar e estruturar a comunicação do utilizador com os seus colegas, oferecendo uma conta de e-mail para tal.

Na fase seguinte, Humanedj para o trabalho, o utilizador usa o software para coordenar e automatizar as suas tarefas de trabalho.

Finalmente, na última fase o Humanedj é utilizado para organizar e gerir o seu trabalho, bem como o dos seus colegas.

Contudo, os criadores desta ferramenta realçam que o software por si só não é suficiente para transformar completamente a vida de trabalho do utilizador. Na figura 5 pode ver-se o ambiente de trabalho do Humanedj.

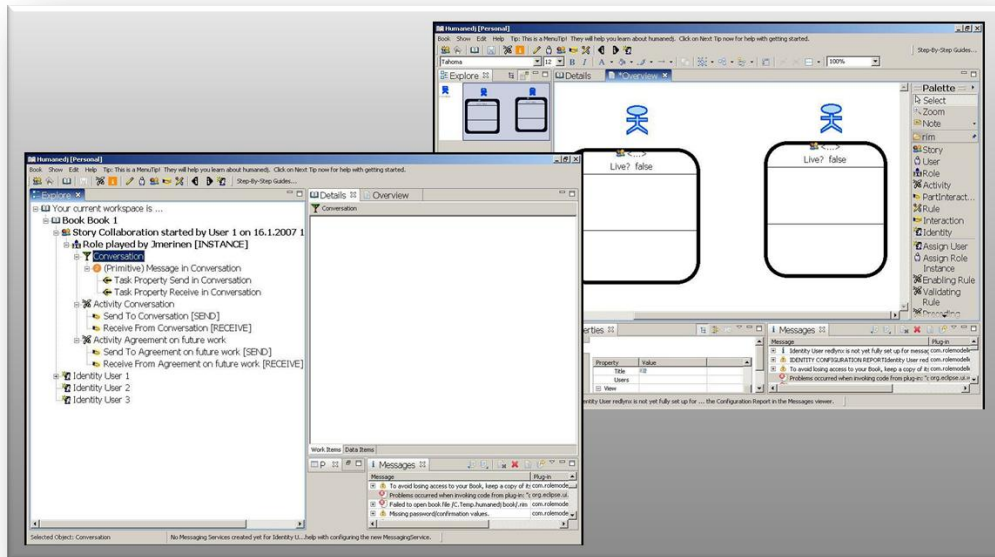


Figura 5 – Interfaces do Humanedj

Na tabela que se segue, está o resultado da comparação do Humanedj com outra ferramenta colaborativa, neste caso o Microsoft Outlook, onde foram avaliados os cinco elementos do HIM, enumerados no ponto 3.3.

Tabela 1 – Comparação entre as ferramentas Humanedj e Outlook

Elemento HIM	Humanedj	Microsoft Outlook
Visibilidade da comunicação	Actualização automática dos diagramas de interacção (separador "Explore" da figura 5) que descrevem a interacção do processo	Funcionalidades colaborativas como delegação e seguimento de tarefas Visão mais mecanicista dos processos As tarefas e os eventos têm uma lista de membros participantes mas as suas relações internas são ignoradas
Mensagens estruturadas	As mensagens são sempre relativas a uma história e estão organizadas por história na interface do utilizador	As mensagens podem ser organizadas por conjuntos de discussões
Suporte para o trabalho mental	Suporte para planear e registar o trabalho mental em diagramas de interacção	O progresso das tarefas pode ser registado pelo utilizador As horas de trabalho mental podem ser marcadas em tarefas separadas
Gestão da actividade compreensiva	Algum suporte através dos diagramas de interacção	
Processos que alteram processos	Possibilidade de actualizar dinamicamente o diagrama de interacção	Actualização automática de tarefas e eventos mas não suporta verdadeiramente a alteração de processos

Como se pode observar, as funcionalidades de ambas as aplicações são ligeiramente diferentes, isto porque as ferramentas comparadas não desempenham propriamente o mesmo papel.

No entanto, o Humanedj oferece já algumas funcionalidades mas ainda se encontra numa fase muito embrionária e longe de poder ser implementada numa organização, precisando para isso de ser melhorada em vários aspectos, desde a sua relação com o e-mail, até ao aspecto visual e seus termos demasiado técnicos. [6]

2.3. Síntese

Apesar de todos estes sistemas de gestão de projectos, alguns bem sofisticados e demasiado complexos, muitos gestores continuam a preferir a utilização do seu e-mail ou de um simples blogue para gerir o projecto pelo qual é responsável. Isto responde claramente que as soluções existentes no mercado, apesar de inúmeras, não conseguem dar resposta às necessidades e expectativas dos gestores de projectos e suas equipas de trabalho.

A utilização deste tipo de ferramentas também permite perceber que é cada vez mais importante a integração de sistemas e, neste caso em concreto, a integração do e-mail na aplicação de gestão de projectos.

Por fim, a complexidade de alguns destes sistemas é tal que os gestores têm dificuldades em começar a utilizar uma ferramenta deste tipo, acabando por desistir, o que revela a cada vez mais latente necessidade de se desenvolverem aplicações intuitivas e fáceis de utilizar.

Capítulo 3

Novos Paradigmas

Este capítulo introduz um conjunto de conceitos e novos paradigmas sobre as formas de trabalho nas organizações e os modos de interação entre as pessoas. Apresenta ainda o conceito de fluxo de estados de um processo e uma visão sobre o conceito de explorador de actividade.

3.1. Contexto

As organizações deparam-se hoje em dia com o crescente problema de lidar com um largo número de requisitos dos clientes sem aumentar o número de funcionários. Uma possível solução passa por encontrar formas de aumentar a eficácia dos seus processos internos. Tipicamente, uma organização tem que trabalhar com um grande número de processos e não há possibilidade de contratar um gestor para supervisionar a execução de cada processo separadamente, pelo que, transferir parte das funções de controlo e coordenação para um sistema de apoio é, aparentemente, uma forma económica e ao mesmo tempo eficaz e fiável de tratar o problema. No entanto, esse sistema deve tomar a função de supervisor quando o processo segue o seu padrão normal mas deve também permitir o controlo manual do processo quando este sofre desvios.

Produzir uma descrição compreensiva dos processos está longe de ser fácil, dado que os procedimentos operacionais raramente se encontram documentados e, quando estão, não contêm todos os detalhes essenciais e necessários.

Deste modo, a única forma de obter informações sobre as actividades dos processos a serem modelados é intervindo nas pessoas envolvidas nestas actividades. Normalmente, as pessoas sabem o que têm de fazer no seu trabalho apesar de, muito provavelmente não saberem nada sobre o modelo formal do processo.

Trabalhar com as pessoas e falar com elas é a única forma de se obter as informações necessárias para a modelação dos processos de negócio, pelo que, é imprescindível nos dias actuais uma interacção humana profunda para organizações que se queiram manter competitivas.

3.2. Business Process Management

Durante a última década, a optimização dos processos de negócio e a sua integração com outros processos dentro da organização, bem como de organizações associadas, tornou-se uma prática habitual das empresas, no sentido de melhorar a eficiência das suas operações.

O Business Process Management System, BPMS, emergiu como um dos principais desenvolvimentos de apoio à compreensão, comunicação e evolução dos sistemas de informação orientados ao processo. Inicialmente, o conceito de Business Process Management, BPM, focava a coordenação dos processos de negócio, examinando a forma como os sistemas interactuavam entre si e de que modo a lógica de processos de negócio poderia estar embebida. Não obstante, as soluções actuais de BPM podem tratar de muitos procedimentos, predeterminados pela interacção entre os sistemas, e ajudar as empresas a organizarem as suas tecnologias de informação com a integração de aplicações.

Há algumas definições de BPM que, na generalidade, introduzem as ideias chave de controlo de processos, melhoria contínua através da análise e implementação de alterações necessárias, reconhecimento das capacidades das pessoas e da importância do ambiente envolvente.

A primeira onda do BPM começou a ser delineada nos anos vinte com a teoria de gestão de Frederick Winslow Taylor¹, onde os processos estavam implícitos nas práticas de trabalho.

Na década de noventa, com a segunda onda do BPM a tendência era para automatizar os processos de negócio manuais, embora sem o sucesso esperado pois normalmente resultavam em sistemas inflexíveis.

Com o aparecimento da terceira onda do BPM, é sugerido que se evite, tanto quanto possível, o desenvolvimento de sistemas orientados ao processo, dando-se mais importância à informação. [6]

Porém, nos dias de hoje assiste-se a uma quarta onda do BPM, que aponta para um novo conceito e um novo paradigma, isto é, para os sistemas de gestão da interacção humana, Huma Interaction Management Systems, HIMS. [7]

¹ Engenheiro Mecânico norte-americano que ficou conhecido pela melhoria da produtividade das pessoas nas indústrias

No início dos anos noventa, os processos eram definidos como um estruturado e avaliado conjunto de actividades para produzir um resultado final específico, durante um período de tempo bem determinado. Isto levou ao desenvolvimento de sistemas de fluxos de trabalho, *workflows*, que eram utilizados como ferramentas para a implementação dos processos. Contudo, estas ferramentas tinham um problema comum pois, na prática, raramente se encontravam aptas a executar o processo para o qual foram modeladas, principalmente quando se tratavam de processos pouco estruturados. Hoje em dia, esse problema continua a ser alvo de vários estudos na tentativa de flexibilizar os fluxos de trabalho, de forma a torná-los mais robustos a processos que não são completamente invariáveis.

Com o objectivo de perceber e gerir os processos de negócio do mundo real, é fundamental perceber e gerir também as interacções humanas e os processos de trabalho orientados às pessoas, *human-driven*.

3.3. Human Interaction Management

Com a evolução das novas tecnologias, há cada vez mais interacções entre as pessoas, tanto em número como em complexidade. Situações banais nos dias de hoje como *outsourcing*, marketing ou outras operações globais aumentaram drasticamente a necessidade de interacção com vendedores e parceiros de negócio. A tendência para interacções cada vez mais complexas têm severo impacto na forma como as empresas se organizam e operam.

Muitos dos sistemas de software não passam de um conjunto de funcionalidades disponíveis através de uma interface, em vez de estarem preparados para responder às necessidades dos clientes e, se possível, exceder as suas expectativas.

De modo a abrir caminho ao desenvolvimento de sistemas altamente interactivos, é necessário passar de uma perspectiva centrada na aplicação para uma perspectiva centrada no utilizador. Hoje em dia, o significado de interacção é cada vez mais abrangente e sofisticado, o que implica a integração de modalidades heterogéneas como a voz, gráficos e animações, bem como a interligação de variados dispositivos como computadores, telemóveis, PDAs, entre outros.

Em suma, os sistemas devem estar aptos para serem utilizados em qualquer contexto, isto é, a adaptarem-se dinamicamente e a criar a interface de acordo com o papel do utilizador e o ambiente em questão, sem necessidade de qualquer alteração na aplicação.

Para se encontrar uma forma eficaz de perceber, modelar, gerir e acompanhar os processos orientados às pessoas, é necessário trabalhar com as pessoas em vez de lhes impor estruturas que, normalmente, são vistas como entraves que dificultam a realização do trabalho. Fazer qualquer coisa que não se ajuste à forma como as pessoas realmente trabalham é simplesmente desperdício de tempo.

A capacidade individual para tomar decisões sobre o que é necessário fazer num dado momento é, normalmente, uma das principais razões pelas quais a pessoa foi escolhida para o lugar que ocupa. Deste modo, deve-se tirar partido desse facto e do profundo conhecimento que a pessoa tem sobre o seu trabalho, dando-lhe a liberdade de decidir o que fazer num dado contexto, em vez de lhe impor uma estrutura rígida à qual teria que se submeter.

Passando para o campo da Qualidade, se tivermos em atenção um dos seus princípios, deve-se confiar nas pessoas e no trabalho que realizam para assim se garantir trabalhadores mais motivados e melhores resultados em termos de eficiência e produtividade.

Contudo, estas decisões devem ser tomadas de uma forma controlada uma vez que a definição do trabalho envolve a determinação de responsabilidades dos seus trabalhadores, obrigações que devem ter e objectivos que devem cumprir.

O HIM é uma nova teoria que descreve como as pessoas realmente trabalham e de que forma se pode ajudá-las a trabalhar melhor, ou seja, a aumentar a sua produtividade. Esta teoria define as mudanças a introduzir nas práticas de trabalho, através da aplicação de princípios que revolucionam os métodos de trabalho em ambientes colaborativos. Estes princípios requerem uma nova abordagem quanto à descrição, gestão, acompanhamento e análise dos processos, pois, apesar das técnicas actuais funcionarem muito bem com actividades mecânicas e repetitivas, estas deixam muito a desejar no que respeita a actividades dinâmicas, inovadoras e interactivas, típicas de comportamentos humanos.

Neste sentido, o HIM mostra como modelar todos os processos de trabalho humano tendo em perspectiva uma resposta apropriada com software e, conseqüentemente, tornar mais acessível a participação e avaliação das pessoas.

As pessoas não são programas informáticos nem máquinas que operam sempre da mesma forma, logo, o seu comportamento não pode ser propriamente descrito, controlado ou facilitado usando técnicas como BPEL, BPMN ou UML.

A abordagem do HIM foi desenvolvida para lidar com comportamentos humanos numa organização, desenvolvendo ideias não só sobre a teoria dos processos humanísticos mas também sobre a teoria da psicologia, educação, biologia e dos sistemas sociais. Jon Pyke² caracteriza o HIM como o mecanismo que converte as estratégias de topo das organizações em actividades colaborativas executáveis, proporcionando também uma aproximação para negociar os processos públicos.

O Human Interaction Management como um todo é uma teoria multi-camadas, descrevendo modelos de processos que podem ser usados para compreender a actividade de trabalho humana, providenciando ferramentas para a modelação dessa mesma actividade, técnicas para a sua gestão e tecnologias o seu acompanhamento.

No centro do HIM encontra-se sempre o conceito de papéis desempenhados pelo participante no processo, isto é, a representação das suas funções no processo, que inclui

² Um dos pioneiros do BPM e o criador do primeiro fluxo de trabalho implementado em JAVA

objectivos, responsabilidades, fontes de informação, etc. No modelo de processo convencional, o conceito de papel limita-se às actividades no grupo, o que seria claramente insuficiente para a modelação das interacções humanas e completamente desajustado para a gestão de um processo continuamente em mudança como é a colaboração humana.

Keith Harrison-Broninski³, o maior impulsionador do HIM, refere que um negócio que queira manter-se competitivo deve adquirir sistemas eficientes para a gestão dos seus processos. Contudo, as ferramentas existentes não são aplicáveis a processos complexos envolvendo interacção humana como elemento central, pois não têm capacidade para capturar elementos humanos cruciais para estes processos.

O mesmo autor considera as técnicas de modelação tradicional de processos, que se aproximam do BPM, como por exemplo BPMN ou BPEL, desenvolvidas especificamente para a descrição de actividades de rotina e altamente automatizadas, apenas com um envolvimento humano ocasional. Quando os processos se tornam conduzidos por humanos, tais técnicas não são mais aplicáveis.

Há cinco funcionalidades importantes e necessárias para que a gestão das interacções humanas resulte de forma eficiente e produza resultados para as organizações.

A primeira é a visibilidade da comunicação pois os processos humanos criam significativas comunicações entre os vários participantes, logo, cada participante deve ter conhecimento dos outros participantes e do que eles fazem, de modo a trabalharem em conjunto de forma eficiente. A representação dos participantes do processo é assim necessária, mostrando os papéis que desempenham e as fontes de informação privada às quais têm acesso.

A segunda corresponde às mensagens estruturadas e controladas pois só assim se consegue tirar partido do valor das mensagens, como por exemplo o e-mail. Por conseguinte, o tratamento e armazenamento destas mensagens são determinantes para um bom desempenho.

O terceiro elemento diz respeito ao suporte para o trabalho mental dado que o esforço mental investido é um trabalho não mensurável e, conseqüentemente, uma parte crítica do trabalho do participante no processo.

O quarto aspecto prende-se com a gestão da actividade que deve ser compreensiva em vez de descritiva, uma vez que as pessoas não são autómatos programáveis que sequenciam as suas actividades. Além disso, as pessoas trabalham de forma diferente em dias diferentes dependendo do seu estado de espírito e de humor, entre outros factores. Deste modo, as pessoas devem poder dar seguimento a qualquer actividade, desde que estejam reunidas as condições para tal, independentemente da actividade que tenha sido finalizada antes. De forma análoga, qualquer actividade que não tenha sido completamente concluída deve ser dada como não executada, para evitar que isso venha a prejudicar o processo completo.

³ Consultor, investigador e programador que trabalha na vanguarda das TI no mundo dos processos de negócio

A última funcionalidade diz respeito aos processos que alteram processos, isto porque os processos orientados às pessoas são permanentemente alterados pelas acções e interacções das mesmas sobre o processo, e as suas evoluções devem poder ser implementadas na gestão diária dos processos.

Para alcançar todas estas funcionalidades, uma nova geração de sistemas de software é necessária, baseada nos seis diferentes tipos de objectos definidos pelo HIM, que são:

- Utilizadores, intervenientes no processo
- Papéis, funções dos utilizadores
- Interacções, entre utilizadores
- Recursos, informações necessárias para o processo
- Estados, do processo
- Actividades, do processo

Através da implementação destes princípios, o HIMS pode não só suportar o trabalho colaborativo, de forma de tornar efectivamente estratégias em acções, como também fornecer mecanismos para assegurar a gestão e o controlo estratégico, executivo e operacional. A implementação destes princípios garante também a gestão da execução do processo em termos de redefinição do próprio processo, garantindo a máxima agilidade e receptividade. [8]

3.4. Getting Things Done

O *Getting Things Done*, GTD, é um método simples e flexível para gerir tarefas e actividades do dia-a-dia, no sentido de maximizar a produtividade pessoal.

O princípio base deste método consiste em libertar a mente das pessoas de compromissos, planos e ideias, entre outras coisas, permitindo que se concentrem nas acções que têm de fazer no momento.

As pessoas organizadas certamente já utilizaram calendários, listas de tarefas, listas de anotações, entre outras ferramentas. No entanto, o que o GTD acrescenta é o método para utilizar todas estas ferramentas sistematicamente em conjunto. Para tal, o GTD proporciona uma compilação de sugestões e ferramentas, organizadas em cinco fases, apresentadas no fluxograma da figura 6, as quais são descritas a seguir.

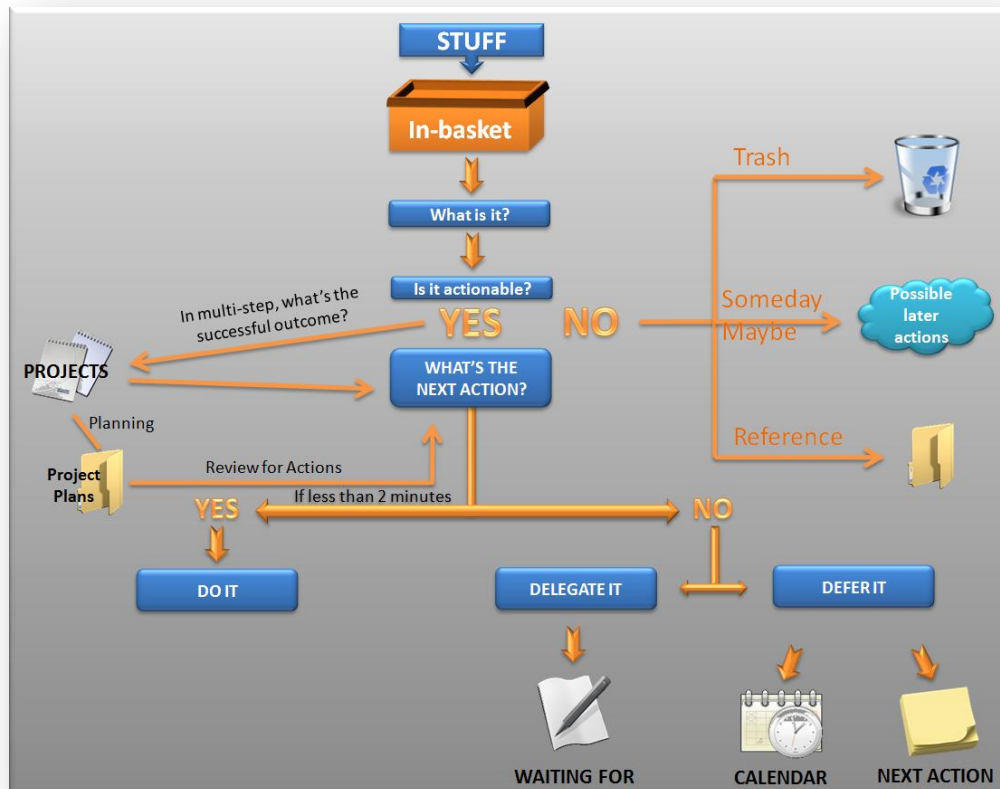


Figura 6 – Fluxograma do *Getting Things Done*

3.4.1. Descrição geral do GTD

1) Recolher elementos que capturem a atenção

A primeira fase trata de acumular tudo que seja potencialmente relevante para a actividade laboral, quer sejam cartas, e-mails, chamadas telefónicas, relatórios, alertas de agenda ou memorandos, entre outros, independentemente do grau de importância ou urgência. Estes elementos estão designados como “*Stuff*” no fluxograma da figura 6.

Para o processo de recolha, são necessárias várias ferramentas desde físicas a electrónicas, como por exemplo pastas de arquivo, bandejas de papéis, aplicações de e-mail, processadores de texto, etc, que todas juntas definem o cesto, definido como “*In-basket*”.

Para não perder o controlo sobre o material acumulado, é necessário esvaziar regularmente o “*In-basket*”, isto é, decidir o que fazer com o quê, tratando desta forma cada item individualmente.

2) Processar os elementos recolhidos quanto ao seu significado e acções necessárias

Para o GTD, processar não é mais do que avaliar cada elemento recolhido e decidir se requer uma acção.

3) Organizar os resultados do processamento

Se não for requerida nenhuma acção, há três possibilidades:

- Eliminar o item se não for necessário
- Colocar na lista de talvez um dia, definido como "*Someday/maybe*", para ser consultado mais tarde, para uma possível acção
- Armazenar o item, classificando-o, para ser consultado quando necessário

Se for requerida uma acção, é necessário decidir o que fazer a seguir:

- Se for necessária mais que uma acção, coloca-se na lista de projectos
- Se a acção não ocupar mais do que dois minutos, não se justifica continuar com a mesma no sistema, por isso, executa-se logo
- Se não for a pessoa indicada para executar a acção, delega-se para a pessoa qualificada para o efeito e cria-se uma nota na lista de espera, designada como "*Waiting*"
- Se a acção tiver que ser executada numa data específica, acrescenta-se mais um evento no calendário
- Se não for possível realizar a acção no momento apenas por questões de tempo, então coloca-se na lista de próximas acções, definido no fluxograma apresentado como "*Next actions*"

Quando se acrescenta um novo projecto à lista de Projectos, deve-se começar por criar um plano do projecto, sem no entanto, ser obrigatório um plano formal com *milestones*, prazos e objectivos específicos, mas apenas com uma visão geral do objectivo do projecto e centrado na lista das próximas acções a fazer, necessárias para avançar com o projecto. Assim que estas acções estiverem definidas, tornam-se parte integrante do fluxograma apresentado, pelo que, é necessário voltar a decidir o que fazer a seguir.

4) Rever as opções para o que estamos a escolher

A fase de revisão é crucial para lembrar o que ainda há para fazer. A revisão diária inclui rever primeiro os eventos do calendário, ou seja, as actividades que têm de ser feitas imperativamente no dia em questão, e depois rever a lista de "*Next actions*", ou seja, o que há para fazer assim que possível.

A revisão semanal é uma revisão mais profunda de todos os elementos que potencialmente requerem uma acção, ou seja, rever o "*In-basket*", o calendário, as próximas acções, os projectos, os planos dos projectos, a lista de espera e a lista "*Someday/maybe*". Isto é fundamental para obter uma visão geral do que há para fazer a curto prazo e verificar quais os prazos que estão a ser cumpridos, dando uma sensação de controlo sobre a situação. Esta revisão semanal inclui também uma limpeza da secretária, do ambiente de trabalho do computador, do e-mail e de outros locais de acumulação de "*Stuff*".

Uma revisão regular é importante para desenvolver e manter a confiança no sistema. Muitas pessoas que não aplicam o GTD, fazem esta revisão poucas vezes por ano, normalmente no início do ano ou quando chegam de férias. Estas boas intenções, contudo, rapidamente se dissipam quando os elementos não processados se começam a acumular e os planos anteriores começam a ficar fora dos prazos ou deixam de fazer sentido por mudança das circunstâncias ou falta de seguimento dos mesmos.

5) Executar

Tendo todas as listas organizadas e dentro dos prazos, o que fazer a seguir? O GTD propõe três modelos para decidir que acção tomar.

1) Modelo dos quatro critérios para escolher as acções no momento

i) Contexto: O que é possível fazer aqui e agora? Por exemplo, não é possível fazer as mesmas acções no escritório ou na rua. Se a lista de próximas acções for extensa, é recomendável que se classifique por contexto, para assim ser mais fácil executá-las juntas.

ii) Tempo disponível: Quanto tempo há disponível agora? Se o tempo for limitado, executam-se apenas as acções de curta duração.

iii) Energia disponível: Que energia resta neste momento? Escolher o que fazer a seguir conforme o nível de energia física e mental. Quando se está cansado, deve-se fazer apenas as acções rotineiras, deixando as acções mais difíceis para quando se sentir mais energético.

iv) Prioridade: Quais são as prioridades? Dado um contexto, um tempo disponível e uma energia suficiente, que acção deve ser feita primeiro? Os dois modelos a seguir respondem a esta questão.

2) Modelo triplo para avaliar o trabalho diário

i) Fazer o trabalho que aparece: Este é o trabalho mais urgente para fazer? Quando se realiza uma tarefa que acabou de aparecer (atender uma chamada telefónica, responder a um e-mail que acabou de chegar, etc) significa que se está a decidir que nesse momento essa tarefa é a mais importante. Alternativamente, se possível, pode-se adiar a execução dessa tarefa e focar-se no trabalho predefinido.

ii) Fazer o trabalho predefinido: Isto significa que sistematicamente se segue a lista das próximas acções.

iii) Definir o trabalho: Se a lista de próximas acções estiver vazia ou se houver a sensação de que a melhor acção a fazer nesse momento é definir o trabalho, então deve-se definir o trabalho. Isto é similar à fase de revisão, onde se limpa o que está na mente e se actualiza o sistema do que há para fazer a seguir.

Para se ter a sensação de que se está a fazer o que é verdadeiramente importante, é ainda necessário aprofundar o conhecimento dos objectivos gerais. Aqui, o modelo a seguir, com analogia à aviação, pode ajudar a clarificar.

3) Modelo dos seis níveis para rever o trabalho

- i) 50.000 pés: Vida.
- ii) 40.000 pés: Visão entre três a cinco anos.
- iii) 30.000 pés: Objectivos entre um a dois anos.
- iv) 20.000 pés: Áreas de responsabilidade.
- v) 10.000 pés: Projectos em curso.
- vi) Pista de descolagem: Acções actuais.

Pode-se definir objectivos para diferentes prazos e espaços de tempo, desde tarefas para se executar imediatamente até planos que se estendem pela vida fora. Este plano que se estende pela vida fora exige que se responda à questão quase filosófica “Qual é o meu objectivo de vida?”. Isto é importante para, de tempos a tempos, engrenar neste pensamento vertical e, rever os objectivos, para não estar constantemente a alterar as prioridades ao nível da pista de descolagem. Como se pode constatar, isto é um método e uma filosofia não só para o trabalho mas para a vida. [9]

3.4.2. Extensão do GTD para apoiar o trabalho colaborativo

O GTD é destinado a tornar o trabalho individual mais fiável através da redução do risco de negligência dos compromissos. Numa organização, se o trabalho individual se tornar mais produtivo, o trabalho do grupo como um todo terá os seus benefícios. Se todas as pessoas, numa organização, se tornarem mais fiáveis na execução das suas tarefas, a organização funcionará de uma forma mais eficiente, lucrando com o aumento da confiança e da sinergia e com a diminuição dos conflitos e da confusão.

Contudo, para além das consequências naturais do GTD numa organização, através do aumento da produtividade pessoal, é possível considerar também contribuições mais directas à eficiência organizacional, através da extensão do GTD ao trabalho colaborativo. Externar informação constitui não só um apoio ao processamento individual de informação mas também uma facilidade na partilha dessa mesma informação, como por exemplo, bibliotecas de documentos, bases de dados, listas de próximas acções e de tarefas comuns, etc, facilitando desta forma a coordenação entre as várias pessoas intervenientes.

Para suportar o trabalho colaborativo é então necessário acrescentar uma nova etapa de decisão no diagrama sobre a organização do fluxo de trabalho sugerida pelo GTD, apresentado na figura 6, que corresponde a decidir onde armazenar o item, ou seja, no sistema de memória

individual ou organizacional. Com esta nova possibilidade, é preciso ter uma análise mais abrangente e cuidada porque, por exemplo no tratamento dos itens que se consideram como lixo, se pertencerem ao sistema organizacional, devem ser arquivados como referência, pois o que aparenta ser irrelevante para uns pode não ser para outras pessoas na organização. [10]

3.5. State-Flow

Um processo de trabalho pode ser visto como um fluxo predefinido de actividades com o objectivo de alcançar um determinado resultado e onde os desvios acontecem raramente. No entanto, em muitos processos, esses desvios são a norma e não a excepção.

Dadas as grandes limitações dos fluxos de trabalho, referidas no ponto 3.1, onde um dos principais problemas é a falta de flexibilidade, o State-Flow, SF, pode ajudar com uma nova visão que considera os processos como uma trajectória no espaço de todos os estados possíveis.

O que distingue a aproximação State-Flow das técnicas de modelação convencionais de processos é a ênfase na flexibilidade do controlo em todos os estados, cuja importância é capital para os processos de escritório.

A definição geral de processos de negócio é baseada na noção de objectivo e a qualquer momento é possível dizer se o objectivo é atingido ou não. Se não tiver sido atingido, deve ser possível saber o quão longe se está do objectivo, ou seja, deve haver uma noção de estado do processo, quer seja intermédio ou final.

Esta noção de estado permite considerar os processos de negócio como um sistema dinâmico que se move no espaço de todos os estados possíveis ate chegar ao objectivo final. O conjunto de actividades que se desenvolvem no espaço de todos os estados possíveis cria um fluxo que tende para o objectivo pretendido, movendo e aproximando desta forma o processo para o resultado final. Apesar de em alguns estados as actividades se afastarem do objectivo final, no global o processo converge para a solução pretendida. A conclusão de um processo não precisa assim de atravessar rígidas sequências de actividades, como é o exemplo dos fluxos de trabalho.

O estado de um processo é representado por uma complexa estrutura que inclui atributos e referências para participantes humanos e não humanos (responsável do processo, documentos, etc). Este estado não apresenta as actividades que têm de ser executadas mas sim que resultados foram obtidos até agora, como por exemplo, um documento que foi redigido e se encontra no estado pendente para aprovação. Deste modo, o estado do processo, e por vezes o seu histórico, é usado como ferramenta principal na decisão sobre o que deve ser feito a seguir para atingir o objectivo do processo no estado actual.

O objectivo pode ser definido como o conjunto de condições que têm de ser satisfeitas, onde o estado que satisfaz o conjunto dessas condições se chama estado final do processo. As actividades planeadas que se seguem devem guardar informações como, por exemplo, tipos de acções, data e hora do planeamento, prazo, nome da pessoa responsável pela acção, etc.

Nos processos de negócio, a progressão pelos vários estados até ao objectivo a atingir é feita através da execução de actividades, de forma automática ou com ajuda humana, onde o estado actual de cada processo normalmente contém informação suficiente para se saber que actividades são necessárias serem executadas a seguir para continuar o processo. Assim que uma actividade terminar, novas actividades podem ser planeadas com base no novo estado do processo, como regras do planeamento dinâmico.

Deste modo, para se formar estas regras, deve-se considerar o planeamento do processo como parte integrante do estado do processo, o que permite definir a noção de estado válido, isto é, para ser válido o estado deve incluir todas as actividades necessárias e permitidas para avançar o processo para o estado seguinte estipulado.

O tratamento do planeamento como parte do estado do processo permite que uma alteração ao plano seja uma operação normal na mudança do estado do processo. O planeamento deve ser estruturado segundo um conjunto de regras baseadas em políticas, tais como, obrigações, proibições e permissões. Contudo, é impossível prever todas as possibilidades que um processo pode tomar a cada instante, pelo que, as regras de planeamento não devem ser rígidas, devendo dar a liberdade para ajustar o plano manualmente, permitindo lidar com os acontecimentos inesperados.

Durante a implementação destas novas filosofias é importante ter uma estratégia gradual de mudança, sendo para tal aconselhável começar sem regras, depois acrescentar algumas recomendações e, numa última fase, transformar essas recomendações em políticas, referidas no parágrafo anterior. [11]

3.6. Activity Centric

O objectivo do Activity Centric, conceito proveniente do AE da IBM, é estabelecer um novo conceito organizacional no ambiente computacional. Esse novo conceito denominado de actividade representa a unidade de trabalho.

Através da criação dos Activity Hubs, que reúnem todas as ferramentas e informações necessárias para trabalhar, os utilizadores trabalharão num ambiente mais completo. Deste modo, através da divisão do trabalho em diferentes actividades, os utilizadores podem se concentrar numa actividade em particular e gerir melhor possíveis interrupções.

O Activity Centric deve permitir aos utilizadores gerirem o trabalho sem os obrigar a navegarem para outra localização.

Os utilizadores têm grandes benefícios se utilizarem uma aplicação com uma interface *self-service*, ou seja, bem desenhada, com um aspecto agradável e que lhes ofereça os serviços que precisam num dado contexto, isto é, que seja intuitiva e fácil de usar. [12]

3.7. Fundamentos Cognitivos do Conhecimento do Trabalho

O conhecimento do trabalho consiste nas várias formas de processamento humano da informação, que inclui actividades como obter informações, interpretar, classificar, resolver o problema e decidir o que fazer.

Esses processos de análise cognitiva baseavam-se, inicialmente, na manipulação de símbolos de acordo com um complexo programa ou algoritmo, assumindo que o conhecimento era uma representação interna e abstracta do exterior. A manipulação destes símbolos permite a execução da tarefa principal da cognição, isto é, resolver problemas, ou seja, responder a questões sobre o ambiente envolvente e estruturar um plano para atingir o objectivo nesse ambiente.

Contudo, o cérebro humano tem as suas limitações, entre as quais se destaca a famosa limitação do mágico número sete. A mente humana não consegue manter em memória e trabalhar mais do que sete elementos em simultâneo, aproximadamente, pois, na realidade, estudos comprovam que este número pode variar entre cinco e nove.

Uma descrição suficientemente detalhada de uma situação real inclui, tipicamente, milhares de símbolos, que podem ser combinados de milhares de formas diferentes. Além disso, o sistema cognitivo está constantemente a interagir com o ambiente exterior, através da percepção dos sentidos e da produção de acções que alteram novamente o ambiente exterior, entrando desta forma num círculo repetitivo.

Deste modo, para libertar a mente de informação, num sistema cognitivo distribuído, pode-se fazê-lo através do seu armazenamento externo, como por exemplo, tomar umas simples notas.

Este sistema de armazenamento externo de memória é mais fiável e consome menos energia, libertando assim a mente para outras acções.

A noção cibernética de controlo por realimentação permite atingir os objectivos por eliminação da diferença entre os mesmos e o estado actual da situação, através de acções tomadas. Contudo, algumas acções aumentam a diferença entre os objectivos e o estado

actual mas, no global, essas acções tendem a diminuir essa diferença. Esta noção de controlo está na base do estado psicológico de fluxo.

Fluxo é o agradável estado que o ser humano experimenta quando está mergulhado numa actividade, na qual sente controlo e se sente capaz de a fazer avançar.

O fluxo é caracterizado pela clara sensação de objectivos e pelo contínuo retorno de informação indicando o quanto a situação está próxima do objectivo. Para se sentir o fluxo, as actividades devem estar em consonância com as capacidades individuais, ou seja, as tarefas não devem ser muito difíceis, para não provocar stress nem ansiedade, nem muito fáceis, para não se tornarem aborrecidas. Durante o fluxo, o ser humano tende a esquecer-se de outros problemas e a perder a noção do tempo, focando-se apenas na tarefa que tem para executar.
[10]

3.8. Aspectos Humanos

O comportamento computacional exibido sob a forma de funcionalidades do sistema é a representação de um objecto.

A semântica é a relação entre a representação e o objecto, estabelecida pelo interpretador.

O desafio em computação é lançado pela subjectividade dessa relação, isto é, dado um objecto, diferentes pessoas podem ter interpretações diferentes e por conseguinte significados diferentes, facto que é mais evidente quando se trabalha num ambiente partilhado ou colaborativo.

Para suprir a subjectividade da semântica, é necessário incorporar os aspectos pragmáticos e sociais no modelo. A figura 7 ilustra como a interacção social determina os processos semânticos e pragmáticos, onde se pode observar que o conjunto de atributos abrangidos pelo significado de um vocábulo, apresentado pelo sistema informático, pode ser interpretado com um conceito diferente dependendo da pessoa que o interpreta.



Figura 7 – Diagrama da subjectividade da semântica

O utilizador final tem dificuldades em perceber, por exemplo, o que está a acontecer no momento actual com as actividades planeadas, pelo que, este espera que o sistema seja intuitivo e claro. A figura 8 apresenta uma solução que dá a sensação de controlo a um utilizador sobre todas as suas actividades.

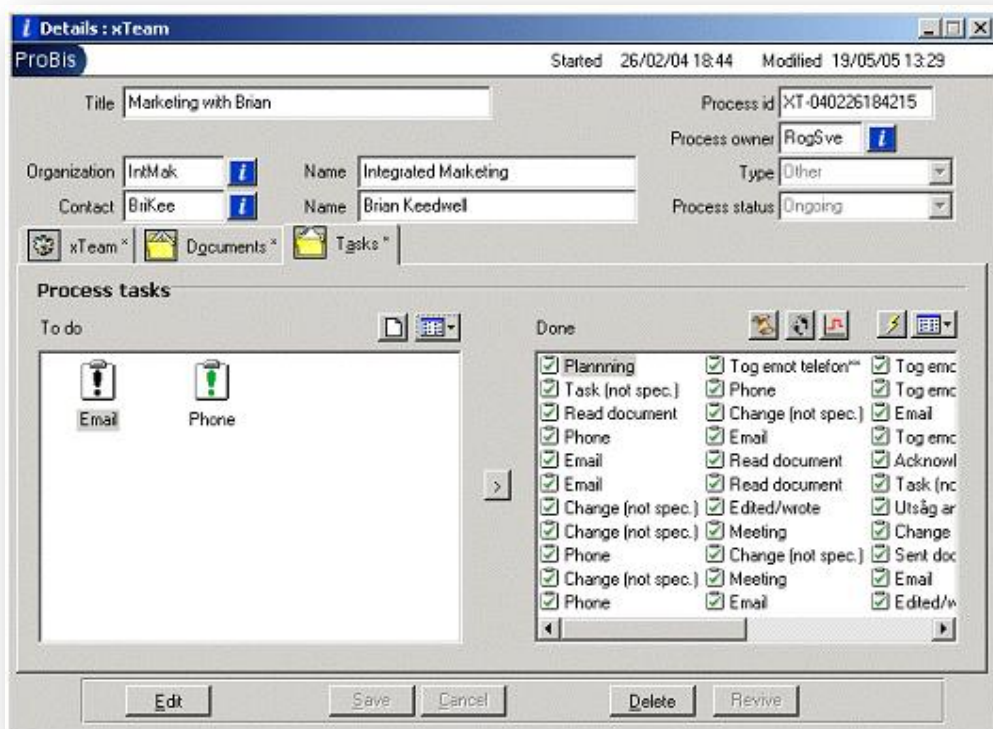


Figura 8 – Interface do ProBis

As funcionalidades que atraem o utilizador são por exemplo, o fácil acesso à informação associada a cada objecto, o aspecto agradável da interface, por exemplo com ícones chamativos e o intuitivo funcionamento da aplicação no seu global. [13]

3.9. Síntese

O BPM assume que os processos podem ser total e antecipadamente planeados e controlados durante a sua execução enquanto que o HIM admite que podem ocorrer imprevistos e o processo pode não ser executado exactamente da mesma forma como planeado.

Embora ambos tentem melhorar os processo, o objectivo do BPM é otimizar as actividades conduzidas pelo processo enquanto que o objectivo do HIM é facilitar e dar suporte à interacção humana no processo. Por outras palavras, BPM está focado no melhoramento dos resultados finais das acções enquanto que o HIM está focado no melhoramento da forma como as acções são conduzidas, isto é, na optimização do ambiente de trabalho e na facilidade de partilha de informação.

Para comparar as abordagens BPM e HIM, Jon Pyke usa uma simples analogia: num jogo de golfe, a abordagem BPM seria acertar no buraco sempre que se desse uma tacada, ou seja, seria apenas necessário dezoito tacadas e aproximadamente vinte e cinco minutos para acabar uma partida. A realidade do golfe é ligeiramente diferente pois um conjunto de situações podem acontecer entre a tacada e a finalização do buraco, e os jogadores têm que lidar com isso. O resultado é um complexo processo com dezoito alvos e setenta e duas operações.

Como tal, apesar de ambos os conceitos terem uma forma de modelar um processo assim como as regras a seguir na sua execução, o BPM adopta uma perspectiva mecanicista dos processos e o HIM uma abordagem mais aberta à consideração de imprevistos e excepções na execução dos processos.

De acordo com os princípios do GTD, as pessoas devem procurar libertar a sua mente de informação desnecessária que possa ser armazenada externamente, para assim poder usufruir e maximizar as suas potencialidades e criatividade. Como se viu, o armazenamento dessa informação deve ser efectuado sob a forma de acções para que seja mais cómodo e rápido interpretar o que há para fazer.

As sensações de stress, ansiedade e excesso de informação devem ser evitadas, substituindo-as pela sensação de controlo, assim como todos os registos de informação efectuados devem estar sob a forma de uma acção, de modo a facilitar a sua interpretação e decisão sobre o que há para fazer a seguir. Deve-se evitar as interrupções frequentes porque o cérebro tem dificuldades em alinhar novamente a sua atenção para a actividade que estava a executar, o que constitui um desperdício dos recursos mentais e uma redução da produtividade do ser humano. Dadas as limitações do cérebro, o seu rendimento é maximizado quando se

recorre a sistemas externos e, neste sentido, esses sistemas devem armazenar informação de forma segura, estimular novas acções e proporcionar retorno de informação sobre as tarefas em curso ou já executadas. Deste modo, proporcionará um complexo conjunto de actividades eficientemente sustentadas e coordenadas que conduzem ao objectivo traçado.

De acordo com a abordagem baseada no fluxo de estados, o cerne do sistema de suporte aos processos consiste num histórico que automaticamente armazena informação de todos os eventos, estados passados, documentos e outros objectos, num planeamento dinâmico, isto é sempre que for preciso, e distribuído, ou seja, não apenas para ele próprio e, por fim, consiste num sistema de navegação que permite navegar livremente no espaço dos processos interligados, no presente e no passado.

Em suma, o BPMS deve permitir o acesso ao estado do processo, em particular à informação sobre o que tem sido feito até agora (estado do processo), como é que tem sido feito (história do processo) e o que é preciso fazer (plano do processo). Isto inclui também informação sobre o quê que cada pessoa deve fazer, sendo necessário para isso promover canais de comunicação eficazes.

O BPMS deve permitir o acesso à experiência da organização, ou seja, a processos já finalizados de modo a ser possível analisar e retirar lições para o futuro.

Como conclusão final, pode afirmar-se que o objectivo da evolução não é desenvolver um sistema que satisfaça as necessidades dos clientes mas sim instruir pessoas para que possam explorar todas as possibilidades desenvolvidas na arquitectura BPMS, o que pode ser considerado como uma espécie de desenvolvimento de um sistema de software ágil, fácil de utilizar, intuitivo e agradável, onde as pessoas se sintam confortáveis, confiantes e moralizadas.

Capítulo 4

Análise Funcional e Especificação

Neste capítulo são abordados e analisados os requisitos do sistema que se pretende implementar, são definidas as entidades para o modelo conceptual do domínio da solução e são especificados os casos de uso que permitem implementar esse modelo.

4.1. Contexto

Tal como referido na Introdução, era pretendido um sistema flexível de gestão de múltiplos projectos, que se adaptasse a vários tipos de projectos, quer sejam projectos simples ou com um grau de complexidade elevado, independentemente do nível de estruturação do mesmo. Nesta visão, fazia parte das expectativas uma área agregadora que servisse para apresentar os últimos acontecimentos dos vários projectos, dando assim uma ideia global desenvolvimento dos mesmos e seus estados actuais.

A concretização deste projecto fazia parte de uma ideia, mais ou menos ténue, de um sistema com uma visão futura, baseado em novos paradigmas e tecnologias, para preencher as necessidades dos gestores de projectos em geral, uma vez que os sistemas actuais não davam uma resposta eficaz a temáticas como a colaboração, personalização, agregação, flexibilidade e agilidade.

Neste capítulo foram analisados os requisitos necessários para um sistema de gestão de projectos, com base numa orientação inicial sobre o que se pretendia e no estudo efectuado sobre gestão de projectos, gestão da interacção humana, fundamentos cognitivos e soluções existentes no mercado.

4.2. Análise de Requisitos

A especificação dos requisitos para esta ferramenta foi fundamentada nos estudos sobre gestão de projectos, gestão da interacção humana e fundamentos cognitivos e baseada nas ideias iniciais, referidas no capítulo 1, e nas soluções existentes no mercado, abordadas no capítulo 2.

Tal como já foi referido, as ideias para este sistema concentravam-se no desenvolvimento de um sistema ágil, que permita o acesso rápido à informação e a cómoda inserção de novos elementos no sistema, e flexível o suficiente para se adaptar a diferentes tipos de projectos e diversas formas de trabalhar das pessoas.

Inicialmente, pretendia-se que suportasse a gestão de projectos de pequena e média complexidade, com possibilidade de evoluir para projectos de grande complexidade, no futuro.

O sistema deve possibilitar a agregação de informação a vários níveis, desde o mais elevado que reúna o essencial dos vários projectos, até ao mais detalhado de cada elemento do projecto. Essa agregação de nível superior não deve ser apenas uma listagem dos projectos em curso mas sim uma apresentação sumária dos últimos acontecimentos de todos os projectos, com hiperligações directas a esses acontecimentos. Deste modo, esta ferramenta que se pretende desenvolver pode denominar-se por Gestão Multi-Projectos, uma vez que este é um dos pontos pelo qual esta se distingue das existentes no mercado.

O sistema deve também ter a capacidade e flexibilidade suficientes para permitir a gestão de uma forma quer estruturada quer desestruturada dos conteúdos dos projectos, sem contudo perder a agilidade e rapidez no acesso a esses mesmos conteúdos.

Dadas as funções de um gestor de projectos, especificadas no ponto 2.1, a ferramenta que o apoia deve oferecer métodos para o planeamento do projecto, organização, coordenação e liderança de equipas de trabalho e controlo do trabalho desenvolvido pelas mesmas na evolução do respectivo projecto.

De igual modo, dadas as funções do gabinete de projectos, uma vez que a ferramenta deve ser colaborativa e não apenas para o trabalho de uma única pessoa, o sistema deve permitir a preparação de planos de actividades, o controlo das mesmas e a respectiva elaboração de relatórios de evolução, deve oferecer mecanismos para a resolução de problemas relativos ao projecto e estar preparado para eventuais alterações do plano inicial.

Deve ainda permitir o arquivo de toda a documentação do projecto, bem como a elaboração de documentos, como por exemplo a elaboração de normas da qualidade ou a redacção de actas, etc, com base em modelos de documentos, previamente definidos, para os vários tipos de documentos habituais num projecto.

O sistema deve oferecer informações, quer em termos numéricos quer percentuais, sobre o tempo disponível até à conclusão prevista do projecto e a quantidade de actividades por realizar, cuja importância se encontra descrita no ponto 2.1. Em termos estatísticos, este deve oferecer também informações sobre o tempo dispendido até ao momento, as variações de

tempo gasto em relação ao planeado e uma estimativa de tempo total que será consumido até à conclusão do projecto, o custo actual do projecto e respectivo saldo restante, as variações de custos em relação ao orçamento e estimativas sobre o custo total do projecto, tal como explicado no mesmo ponto 2.1. No que respeita a indicadores estatísticos sobre a qualidade, devem ser apresentadas informações sobre actividades que não atingiram o pretendido ou que não tenham sido concluídas, e respectivas frequências de ocorrência, e número de correcções que tenham sido necessárias.

A integração com o e-mail e a relação entre todos os elementos do sistema é considerado fundamental para a gestão do trabalho hoje em dia, dada a importância e elevada utilização que o e-mail tem na vida das pessoas.

Tal como referido no capítulo 3, os sistemas devem estar aptos a adaptarem-se dinamicamente de acordo com o papel do utilizador e o ambiente em questão, pelo que, uma gestão eficaz de permissões e utilizadores e uma gestão da personalização do ambiente de trabalho de cada utilizador são também pontos importantes a considerar como requisitos.

Segundo as directivas do HIM descritas no ponto 3.3, para que um sistema destes suporte a colaboração de forma eficiente, deve permitir a visibilidade da comunicação, para que os utilizadores tenham conhecimentos dos utilizadores, bem como das suas funções e permissões, e o controlo e estruturação de mensagens. Deve ainda dar suporte ao trabalho mental, facilitar a gestão da actividade para que esta seja compreensiva e aceitar processos que alteram processos. De notar que, para suportar estas funcionalidades, o HIM sugere seis tipos de objectos que são, Utilizadores, Papéis, Interações, Recursos, Estados e Actividades, tal como enumerados no mesmo ponto.

Seguindo as sugestões do GTD, descrito no ponto 3.4, e de modo a libertar a mente das pessoas, este sistema deve oferecer mecanismos não só para o armazenamento de informação mas também para o armazenamento de informação sob a forma de acções, auxiliado por um sistema de alertas conveniente.

De acordo com o State-Flow, descrito no ponto 3.5, o conjunto das actividades realizadas pelas pessoas faz o processo convergir de estado em estado até ao objectivo final. Deste modo, o sistema deve incluir atributos que definam o estado do processo, bem como o histórico do mesmo.

Seguindo a ideia do Activity Centric da IBM, explicada no ponto 3.6, a interface do sistema deve ser bem desenhada, com um aspecto agradável, intuitiva e fácil de usar. O número de elementos apresentados nesta interface não deve contudo estar muito longe dos sete, de acordo com os fundamentos cognitivos apresentados no ponto 3.7.

4.3. Modelo Conceptual do Domínio

Para dar resposta a estes requisitos, são necessárias várias entidades, que agrupam vários objectos, para a construção de um modelo a implementar no sistema.

Tabela 2 – Tabela de Entidades, Objectos e correspondência com os objectos de HIM

Entidades	Objectos	Correspondência com Objectos de HIM
Documento	Legislação Orçamentos Contratos Relatórios Auditorias Actas Faxes	Recursos Recursos Recursos Recursos Recursos Recursos Recursos
Registo	Anotações	Estados
Comunicação	E-mails Mensagens Problemas	Interacções Interacções Interacções
Actividade	Tarefas Compromissos Reuniões Alertas	Actividades Actividades Actividades Actividades
Informação	Projectos Fases Ficha do Projecto Contactos Pesquisa	Recursos Recursos Recursos Recursos Recursos
Utilizador	Utilizadores Grupos Permissões Funções	Utilizadores Utilizadores Papéis Papéis

Com base nestas entidades, e para que o sistema seja de facto ágil e flexível para permitir relacionar todos estes objectos, é necessário um modelo que suporte todas estas relações, mesmo que algumas não façam sentido, aparentemente, pelo que, um protótipo deste modelo seria o representado na figura 9.



Figura 9 – Modelo Conceptual do Domínio

4.4. Casos de Uso

A partir deste modelo, os casos de uso que permitem implementar as funcionalidades necessárias para o pôr em prática são os descritos a seguir, cuja árvore respectiva é apresentada em primeiro lugar.

Gestão Multi-Projectos

- Gestão documental
 - Inserir documentos
 - Consultar documentos
 - Editar propriedades dos documentos
 - Reorganizar documentos
 - Eliminar documentos
 - Recuperar documentos
 - Aprovar documentos

- Gestão colaborativa
 - Editar documentos
 - Inserir anotações
 - Consultar anotações
 - Editar anotações
 - Eliminar anotações
 - Gestão da comunicação
 - Criar memorandos pessoais
 - Editar memorandos pessoais
 - Eliminar memorandos pessoais
 - Enviar mensagens para
 - Ler mensagens
 - Eliminar mensagens recebidas
 - Guardar e-mails
 - Consultar e-mails
 - Eliminar e-mails
 - Comunicar problemas
 - Consultar problemas
 - Editar problemas
 - Eliminar problemas

- Gestão da informação
 - Pesquisar
 - Inserir ficha do projecto
 - Preencher ficha do projecto
 - Editar ficha do projecto
 - Consultar ficha do projecto e indicadores estatísticos
 - Eliminar ficha do projecto

- Gestão de utilizadores
 - Criar grupos de utilizadores
 - Adicionar utilizadores
 - Consultar grupos e utilizadores
 - Editar grupos de utilizadores
 - Remover grupos de utilizadores
 - Remover utilizadores do grupo

- Gestão do projecto
 - Criar projectos
 - Consultar projectos
 - Editar projectos
 - Eliminar projectos
 - Arquivar projectos
 - Criar fases
 - Consultar fases
 - Editar fases
 - Eliminar fases
 - Criar eventos
 - Consultar eventos
 - Editar eventos
 - Eliminar eventos
 - Adicionar contactos
 - Consultar contactos
 - Editar contactos
 - Eliminar contactos

- Gestão da personalização
 - Personalizar o seu ambiente de trabalho

O sistema de gestão multi-projecto reúne várias funcionalidades que se podem agrupar em seis pacotes de casos de uso, tal como se apresenta na figura 10.

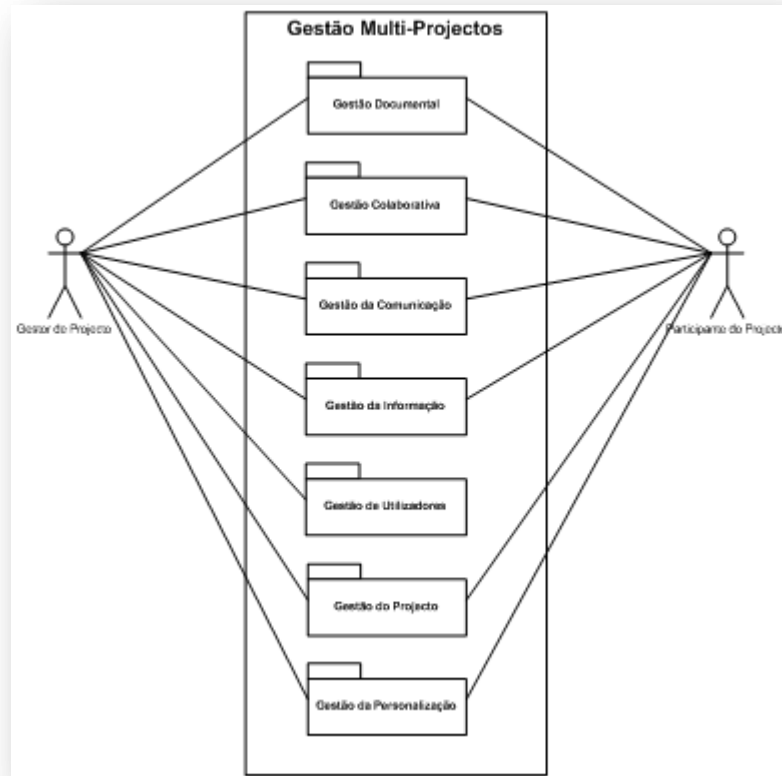


Figura 10 – Pacote de Casos de Uso da Gestão Multi-Projectos

Os actores intervenientes neste sistema são o Gestor de Projecto, com acesso total a todas as funcionalidades do sistema, e o Participante do Projecto que apenas tem acesso a algumas funcionalidades.

De seguida, são apresentados os vários casos de uso dos diferentes pacotes.

4.4.1. Gestão Documental

Inserir documentos

Para inserir documentos para o sistema, o utilizador pode fazê-lo por *drag-and-drop* ou carregando os ficheiros para o sistema através de *upload*. Esse *upload* pode ser de apenas um ou de vários ficheiros ao mesmo tempo.

Consultar documentos

A consulta de documentos permite visualizar todos os documentos bem como apenas alguns, dependendo do critério de filtragem. O aspecto desta visualização pode ser em forma de lista ou idêntico ao explorador de ficheiros do Windows

Editar propriedades dos documentos

É possível editar as propriedades do documento, como por exemplo o seu nome ou outras informações relativas ao documento

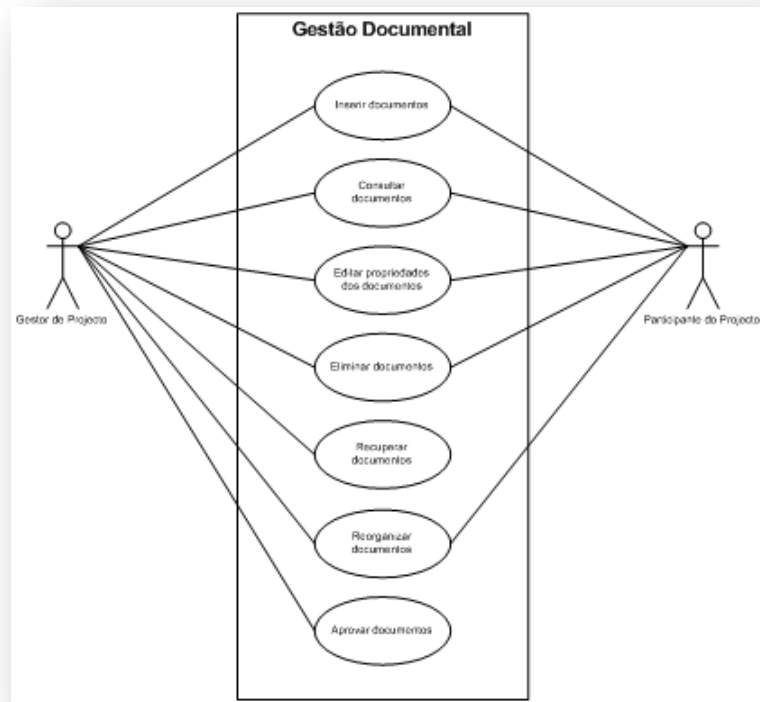


Figura 11 – Casos de Uso da Gestão Documental

Eliminar documentos

Existe a possibilidade de eliminar documentos.

Recuperar documentos

Existe também a possibilidade de recuperar documentos eliminados, dentro de um certo prazo.

Reorganizar documentos

A reorganização documentos pode ser feita através do explorador de ficheiros.

Aprovar documentos

Existe a possibilidade de ter documentos sujeitos a uma prévia aprovação.

4.4.2. Gestão Colaborativa

Editar documentos

Diferentes utilizadores podem editar o mesmo documento e, se necessário, submetê-lo para aprovação superior.

Inserir anotações

É possível inserir anotações sobre o desenvolvimento do projecto e tarefas concluídas, por forma a ficar como registo histórico do mesmo

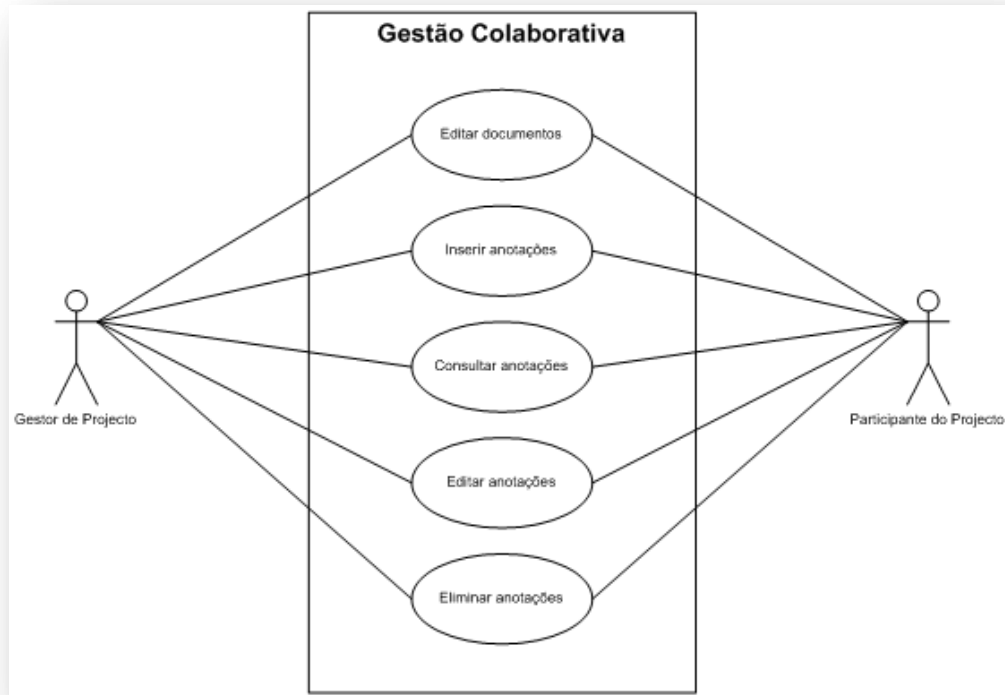


Figura 12 – Casos de Usos da Gestão Colaborativa

.Editar anotações

A edição de anotações só é possível pelo seu autor.

Eliminar anotações

O utilizador pode eliminar apenas as anotações inseridas por si. Estas também poderão ser eliminadas por outros utilizadores que tenham permissões para tal, por exemplo algum superior hierárquico.

4.4.3. Gestão da Comunicação

Criar memorandos pessoais

É possível criar uma nota pessoal com a possibilidade de definir uma data e hora de alerta. Na altura do alerta, é possível definir uma data e hora posterior para a repetição do alerta.

Editar memorandos pessoais

A edição de memorandos permite alterar não só o texto em si como também a data e hora de alerta.

Eliminar memorandos pessoais

É possível eliminar memorandos a qualquer altura, independentemente do alerta ter ou não ocorrido. Na altura do alerta, há a possibilidade de eliminar logo os memorandos.

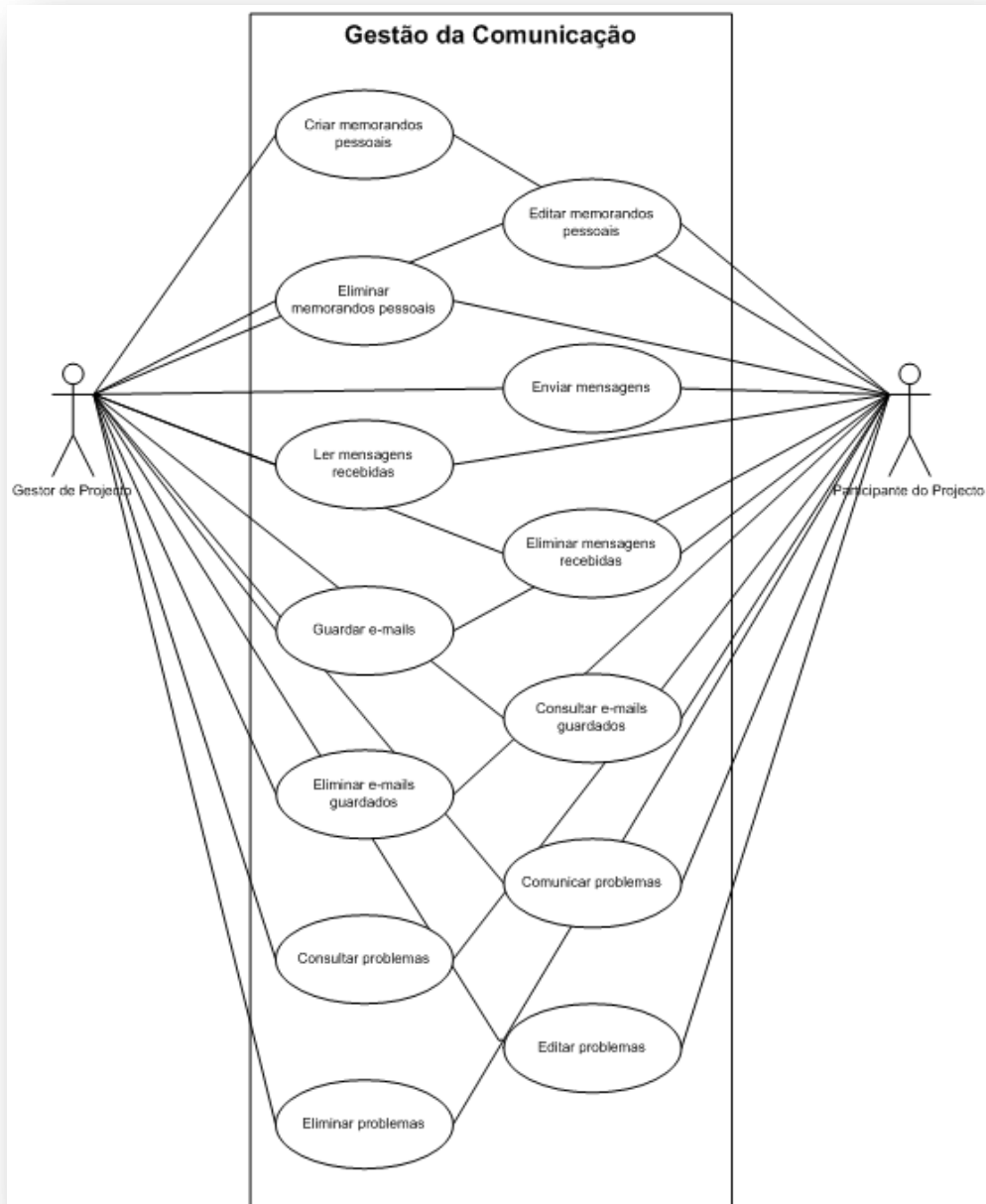


Figura 13 – Casos de Uso da Gestão da Colaboração

Enviar mensagens

Os utilizadores podem enviar mensagens para um ou mais utilizadores que estejam envolvidos no âmbito do projecto e tenham autorização para aceder ao sistema.

Ler mensagens recebidas

Os utilizadores podem igualmente receber mensagens de outros utilizadores, com a possibilidade de filtrar as mensagens recebidas e ordenar por remetente, data e hora.

Eliminar mensagens recebidas

Cada utilizador pode eliminar as suas mensagens recebidas.

Guardar e-mails

É possível guardar e-mails directamente do cliente de e-mail o sistema de gestão multi-projectos.

Consultar e-mails guardados

A consulta de e-mails guardados, à semelhança das mensagens, também permite filtrar e ordenar os e-mails recebidos.

Eliminar e-mails guardados

Cada utilizador com permissões para tal pode eliminar os e-mails guardados.

Comunicar problemas

Quando ocorre algum problema no projecto, este deve ser comunicado especificando a descrição do mesmo, as implicações que acarreta e os impactos causados, podendo existir ainda uma proposta para uma acção correctiva.

Consultar problemas

É possível consultar os todos problemas ou apenas alguns segundo critérios de filtragem e ordenação.

Editar problemas

Quando um problema é lido e é tomada alguma acção correctiva, pode-se alterar o estado desse mesmo problema. É possível também editar outros campos do problema.

Eliminar problemas

A eliminação de problemas só pode ser feita pelo seu autor ou algum superior hierárquico.

4.4.4. Gestão da Informação

Pesquisar

Os utilizadores podem pesquisar em qualquer lugar qualquer elemento do sistema, quer seja um projecto, uma fase do projecto, um evento, um documento, uma anotação, um e-mail, um contacto, um memorando ou uma mensagem. De referir que os resultados da pesquisa são limitados às permissões de cada utilizador, isto é, se o utilizador não tiver permissões para ver um determinado documento, este não aparecerá.

Inserir ficha do projecto

Se a ficha do projecto for um documento, esta poderá ser inserida num local específico para tal.

Preencher ficha do projecto

Se não existir ficha do projecto, esta poderá ser criada através do preenchimento de um formulário do sistema. No entanto, mesmo que haja uma ficha do projecto em formato de documento, alguns campos do referido formulário podem ser preenchidos por forma a introduzir dados para serem tratados e apresentados estatisticamente.

Editar ficha do projecto

A ficha do projecto pode ser editada, caso esta tenha sido criada no sistema, para qualquer alteração ou para a introdução e actualização de dados necessários para os indicadores estatísticos.

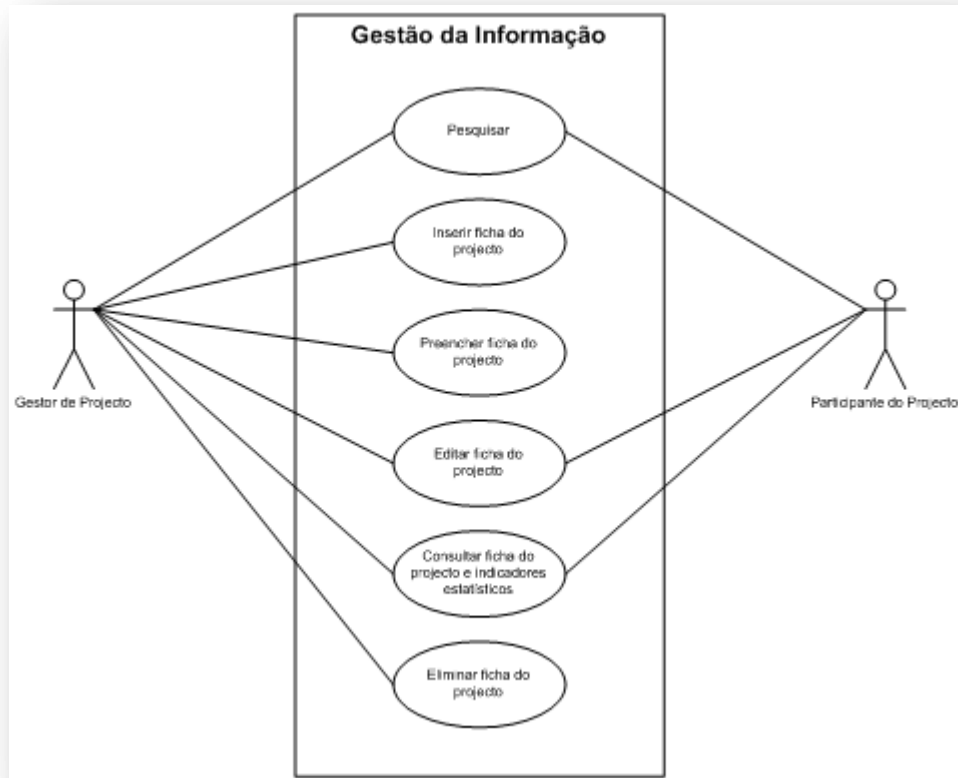


Figura 14 – Casos de Uso da Gestão da Informação

Consultar ficha do projecto e indicadores estatísticos

Existe uma área específica para apresentar os dados da ficha do projecto, quer tenha sido inserida como um documento ou criada no sistema. Nesta mesma área é possível consultar alguns indicadores estatísticos relativos ao projecto, tais como horas dispendidas, custos e saldo actuais do projecto face ao orçamento previsto, atrasos temporais do projecto e das suas fases.

Eliminar ficha do projecto

É possível eliminar a ficha do projecto bem como todos os dados introduzidos para os indicadores estatísticos.

4.4.5. Gestão de Utilizadores

Criar grupos de utilizadores

Quando se cria um novo grupo, pode-se definir o proprietário do mesmo e todas as permissões possíveis para o grupo. É possível também configurar o grupo quanto à visibilidade dos seus membros, apenas visível para os membros do grupo ou para todos os utilizadores, quanto à de edição dos mesmos, apenas possível apenas pelo proprietário do grupo ou também pelos seus membros, e quanto à aceitação de pedidos de associação, pedidos esses que podem ser aceites automaticamente ou não.

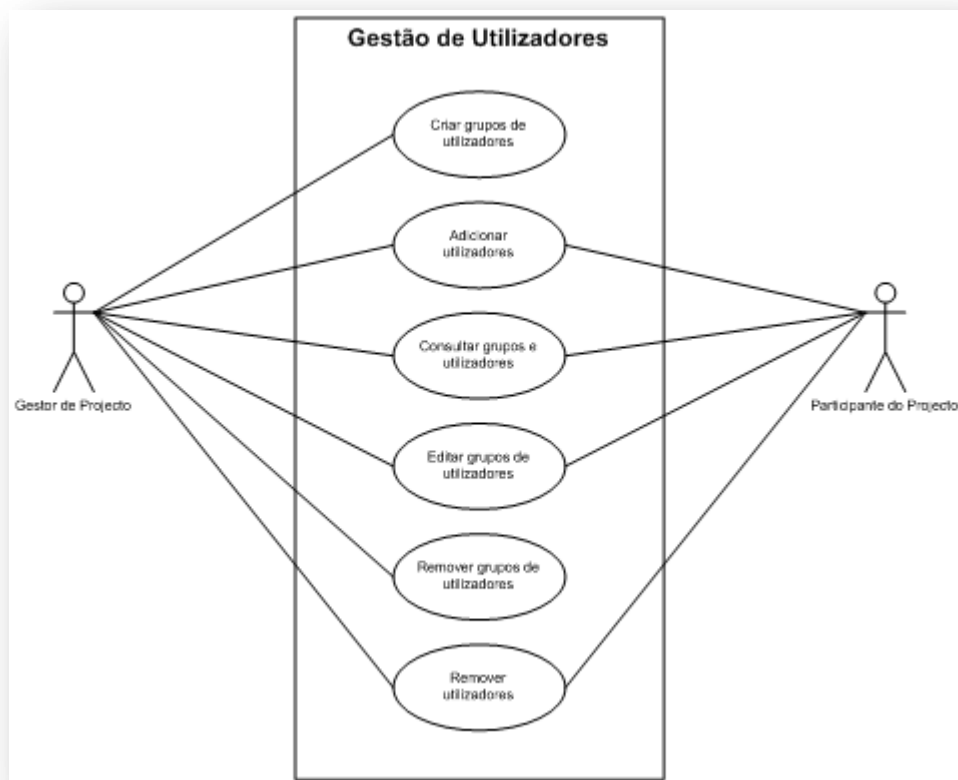


Figura 15 – Casos de Uso da Gestão de Utilizadores

Adicionar utilizadores

É possível adicionar utilizadores, associando-os ou não a um ou mais grupos. Se um utilizador não estiver inserido em nenhum grupo, ele mesmo funciona como um grupo, tendo permissões exclusivas.

Consultar grupos e utilizadores

É possível visualizar os grupos existentes no sistema, as suas permissões e o os membros que os compõem.

Editar grupos de utilizadores

Os utilizadores podem editar os grupos de utilizadores e as suas permissões.

Remover grupos de utilizadores

É possível eliminar um grupo completo de utilizadores.

Remover utilizadores do grupo

É também possível remover utilizadores, quer estejam ou não inseridos nalgum grupo.

4.4.6. Gestão do Projecto

Criar projectos

É possível criar novos projectos com base num modelo que já inclui todos os mecanismos necessários para as funcionalidades descritas.

A criação do projecto é independente do tipo de projecto pois todos os mecanismos encontram-se sempre disponíveis para serem utilizados.

Consultar projectos

É possível consultar os projectos existentes, quer estejam activos, suspensos ou concluídos, bem como ter uma visão agregada com as últimas actualizações e desenvolvimentos dos vários projectos em que os utilizadores se encontram inseridos.

Editar Projectos

Existe a possibilidade de alterar o nome do projecto.

Arquivar projectos

Quando um projecto é concluído, ou fica suspenso, o arquivo permite editar o estado do projecto para assim organizar melhor a apresentação dos projectos activos.

Eliminar projectos

Pode-se eliminar projectos cujo arquivo não se justifique ou tenha sido criado por engano.

Criar fases

Dentro de um projecto é possível criar novas fases. No entanto, esta criação não é obrigatória, como por exemplo no caso de projectos simples.

Consultar fases

A consulta de uma fase permite ver todos os elementos, como por exemplo anotações, tarefas ou documentos, aos quais essa fase está associada.

Editar fases

É possível alterar o nome da fase.

Eliminar fases

É possível também eliminar uma fase, embora isso implique a eliminação de todos os objectos à qual estão associados.

Criar eventos

A criação de eventos engloba a criação de todos os itens de calendário, quer sejam tarefas, compromissos, reuniões ou outros acontecimentos. Como tal, um evento pressupõe no mínimo uma data e hora. Para além disso, é possível definir o evento para todo o dia, a sua periodicidade, a sua prioridade, o seu estado e a pessoa ou grupo a quem se destina, entre

outras propriedades. Se por exemplo um utilizador criar um evento destinado a si próprio, este só aparecerá para si, servindo por isso como uma agenda pessoal também.

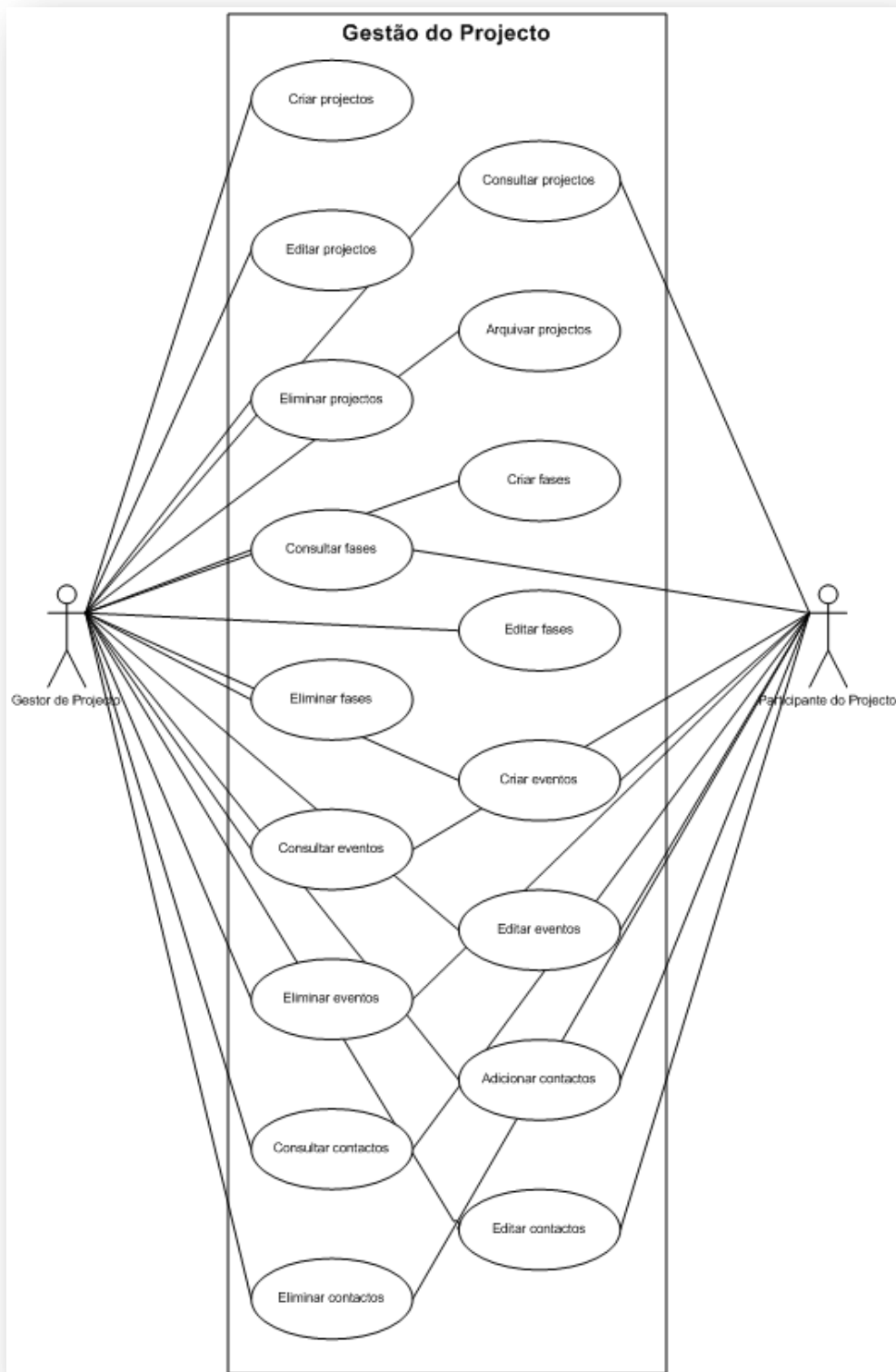


Figura 16 – Casos de Uso da Gestão do Projecto

Consultar eventos

Os eventos criados podem ser apresentados sob vários critérios, como por exemplo, eventos criados pelo utilizador que se encontra autenticado, eventos destinados a esse mesmo utilizador, eventos da fase seleccionada, etc.

Editar eventos

É possível editar um evento em particular ou, se este estiver inserido numa série periódica de eventos, alterar essa mesma série.

Eliminar eventos

É possível eliminar apenas os eventos criados pelo próprio utilizador.

Adicionar contactos

Existe a possibilidade de adicionar contactos que sejam relevantes para o projecto, o que faz com que essa gestão de contactos seja parte integrante do sistema. Aqui, um contacto pode ser uma pessoa individual como uma empresa, onde se pode guardar, entre outras coisas, o e-mail, o telefone, o telemóvel, o número de fax, a morada ou a página Web.

Consultar contactos

É possível consultar os contactos adicionados e ver todos os seus detalhes.

Editar contactos

Os utilizadores podem alterar e actualizar as informações dos contactos.

Eliminar contactos

É possível também eliminar contactos que já não sejam necessários.

4.4.7. Gestão da Personalização



Figura 17 – Casos de Uso da Gestão da Personalização

Personalizar o seu ambiente de trabalho

Os utilizadores podem personalizar a apresentação e disposição dos seus ambientes de trabalho pessoais. Isto inclui, por exemplo, alterar a localização dos elementos que compõem o ambiente de trabalho, oculta-los ou torna-los visíveis, alterar o tamanho que ocupam no ecrã,

definir o número de itens que pretendem que sejam sempre apresentados sobre um determinado elemento (anotações, eventos ou outros elementos), etc.

Todos os elementos apresentados nestes casos de uso podem ser relacionados entre si, desde que essa relação faça sentido, ou seja, por exemplo, é possível associar um documento a um e-mail, uma anotação a um evento ou um contacto a uma fase, entre outras relações.

4.5. Síntese

Neste capítulo foram analisados os requisitos necessários para um sistema de gestão de projectos, com base numa orientação inicial sobre o que se pretendia e no estudo efectuado sobre gestão de projectos, gestão da interacção humana, fundamentos cognitivos e soluções existentes no mercado.

Em suma, foram criadas seis entidades para suportar um modelo capaz de dar resposta aos requisitos apresentados, sendo essas entidades: Documento, Registo, Comunicação, Actividade, Informação e Utilizador.

Por fim, foram apresentados vários casos de uso, agrupados em sete áreas, ou pacotes de casos de uso, que permitem implementar o modelo criado. As áreas abrangidas pelos casos de uso são: Gestão Documental, Gestão Colaborativa, Gestão da Comunicação, Gestão da Informação, Gestão de Utilizadores, Gestão do Projecto e Gestão da Personalização.

Capítulo 5

Estudo e Análise do SharePoint

Tal como referido nos objectivos descritos na Introdução, era um objectivo da dissertação a documentação de todo o conhecimento adquirido relativamente à ferramenta SharePoint. Esta documentação tem o intuito de assegurar a continuidade da aplicação dos STM, permitir a manutenção de outras aplicações SharePoint e ainda servir de ponto de partida para novos projectos baseados na mesma ferramenta.

O conteúdo aqui apresentado resultou do trabalho conjunto com o colega João Miguel Gomes e Silva no âmbito da sua tese de dissertação: Sistema de Gestão de Equipamentos e Manutenção. Notar que os vários assuntos aqui documentados mereceram um esforço equitativo de ambos e sem esta participação não teria sido possível atingir o conhecimento actual da ferramenta, principalmente pela interdependência dos temas e disponibilidade temporal mútua.

Este capítulo apresenta uma visão geral sobre o SharePoint e uma descrição sumária da estrutura e conteúdos básicos de um portal em SharePoint.

Desenvolvimentos mais detalhados sobre soluções mais práticas encontram-se no Anexo A. Porém, esses detalhes mais técnicos são de extrema importância para o desenvolvimento de soluções personalizadas e ocultam o árduo trabalho inerente à sua pesquisa e exploração através de múltiplos testes efectuados. Muitas das soluções presentes nos Anexo A, não se encontram em livros sobre o tema mas sim em fóruns da comunidade Microsoft.

5.1. Visão Geral

O Microsoft Office SharePoint Server 2007, vulgo SharePoint, é uma plataforma de produtividade com um conjunto integrado de aplicações, com capacidades de servidor, que pode ajudar a eficácia das empresas. Este fornece uma gestão de conteúdos abrangente e uma funcionalidade de pesquisa eficaz, de modo a facilitar a partilha de informações dentro e

fora de uma organização, acelerando processos de negócio partilhados e permitindo uma análise dos dados de negócio mais vasta.

O MOSS 2007 suporta todas as aplicações de intranet, extranet e Web, de uma organização, através de uma única plataforma integrada, em vez de depender de diversos sistemas fragmentados. Além disso, este servidor de gestão de conteúdos e de colaboração fornece aos profissionais de tecnologias da informação, TI, e aos programadores a plataforma e ferramentas de que necessitam para administração do servidor, capacidade de adaptação e interoperabilidade entre aplicações.

Esta plataforma possui uma forte integração com aplicações cliente de ambiente de trabalho, correio electrónico e browsers familiares, de modo a proporcionar uma experiência de utilizador consistente que simplifique o modo como as pessoas interagem com conteúdos, processos e dados de negócio.

No MOSS são identificáveis seis principais áreas funcionais:

Colaboração

Ajuda a manter equipas de trabalho ligadas e produtivas fornecendo acesso fácil a pessoas, documentos e informações de que os utilizadores necessitam para tomar decisões. Aplicações SP incluem colaboração e convergência, capacidades para a gestão do ciclo de vida de documentos, de modificações, de notificações de tarefas, de *Really Simple Syndication*, RSS, e uma interface de utilizador com navegação baseada na Web.

Portal

Os componentes de portal do MOSS 2007 incluem funcionalidades especialmente úteis para estruturação, implementação e gestão de portais da intranet empresarial, Web sites de empresas com acesso à Internet e sites de portal com divisões.

Pesquisa

Fornecem uma experiência de pesquisa consistente com uma interface simples mas poderosa para o utilizador. Apresenta funcionalidades para pesquisa de pessoas, competências e propriedades arbitrárias do conteúdo tais como URL, tipo de documento, autor, etc. Os resultados de pesquisas podem ser facilmente filtrados e ordenados. Funcionalidades, como a correcção ortográfica, sugestões de pesquisa e alertas optimizam a relevância dos resultados.

Gestão de Conteúdos

O Windows SharePoint Services, WSS, fornece várias funcionalidades de gestão de documentos, tais como, controlo de versões principais e secundárias, bloqueio da disponibilização e reserva de documentos, meta-dados, fluxos de trabalho associados a documentos, políticas baseadas no tipo de conteúdo, auditorias e controlos de acesso

baseados em funções ao nível da biblioteca de documentos, da pasta de documentos e de documentos individuais.

A forte integração do Office SharePoint Server 2007 com aplicações da família Microsoft Office comuns, como, por exemplo, Word, Excel, PowerPoint, Outlook, etc, ou outras, como por exemplo, SAP, facilita aos utilizadores a consciencialização e o cumprimento dos requisitos normativos.

O MOSS 2007 baseia-se nestas capacidades para fornecer uma solução de criação de conteúdos melhorada, processamento de documentos de negócio, gestão e publicação de conteúdos Web, gestão de registos, gestão de políticas e suporte de publicações multilingue concebido para manter uma relação entre a versão original e diferentes traduções de um documento.

Os fluxos de trabalho existentes no pacote MOSS 2007 permitem iniciar, controlar e gerar relatórios sobre actividades de negócio comuns, como por exemplo, revisão e aprovação de documentos, controlo de problemas e recolha de assinaturas.

É possível efectuar todas estas actividades sem recorrer ao desenvolvimento de código específico. A modificação e expansão destes processos de fluxo de trabalho de utilização imediata são facilitadas através de ferramentas como o Microsoft Office SharePoint Designer 2007, o sucessor do Microsoft Office FrontPage.

Processo de Negócio Orientado por Formulários

Simplifica os processos de negócio através de formulários electrónicos inteligentes, fáceis de utilizar e baseados em XML, que se integram facilmente com os sistemas existentes. Esta plataforma de cliente / servidor, com segurança avançada, fornece um método rápido de criação e implementação de soluções, centraliza a gestão e manutenção de formulários e ajuda a expandir os processos de negócio a clientes, parceiros e fornecedores.

Business Intelligence

Fornecer capacidades de *Business Intelligence*, BI, para que os utilizadores possam partilhar, controlar, reutilizar e aceder em tempo real, a partir de um browser, a informações para a tomada de melhores decisões. As funcionalidades de BI do MOSS 2007 fornecem o acesso Web e programático às folhas de cálculo do Excel publicadas e reutilização programática de dados, permitindo incorporar indicadores chave de desempenho (*Key Performance Indicators* - KPI), avançados e associados a dados de folhas de cálculo publicadas.

5.2. Arquitectura

A base de funcionamento do MOSS 2007 assenta na tecnologia .NET Framework 3.0 e no motor de fluxo de trabalho Windows Workflow Foundation, WWF. Este software requer um

servidor com sistema operativo Windows Server 2003 com IIS 6.0, servindo-se de uma base de dados Microsoft SQL Server 2005, tal como se pode ver na figura 18. Notar que o MOSS 2007 opera igualmente com as mais recentes versões de todos os requisitos de software já referidos, isto é, Windows Server 2008, IIS 7.0 e Microsoft SQL Server 2008. Para além desta infra-estrutura base, é possível integrar o MOSS com outros componentes adicionais, tais como, *Active Directory* - AD, *Domain Name System* - DNS, servidores de e-mail, etc.

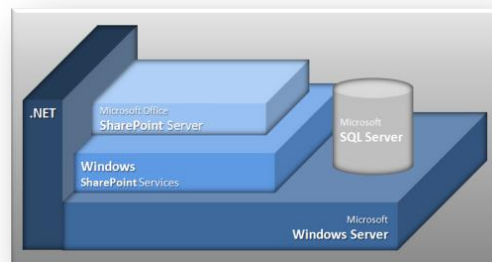


Figura 18 – Arquitectura do SharePoint

O MOSS 2007 é um produto que utiliza a tecnologia WSS, disponibilizado como parte integrante do Windows Server 2003, para fornecer uma estrutura consistente e familiar para listas e bibliotecas de documentos bem como para administração e personalização de portais. No entanto, este produto oferece funcionalidades melhoradas ou adicionais que não se encontram disponíveis num portal do WSS. Por exemplo, o MOSS 2007 utiliza a mesma tecnologia de pesquisa que o WSS, mas inclui funcionalidades adicionais que são especialmente úteis como, por exemplo, a capacidade para pesquisar dados em aplicações externas, quer sejam Microsoft ou não.

5.3. Estrutura de um Portal em SharePoint

Estrutura e Hierarquia de Sites

A estrutura de um portal de SP segue uma hierarquia que se inicia num site de topo, *Top Site Collection*, que agrupa vários sub-sites, *Child Sites*, páginas, listas e bibliotecas, tal como se pode observar pela figura 19.

O site de topo é único, por cada *Web Application* (descrito no Anexo A), podendo, no entanto, agregar várias colecções de sites, *Site Collections*. Uma das diferenças do *Site Collection* em relação aos sub-sites está no facto do primeiro incluir bibliotecas específicas da colecção de sites, bibliotecas essas que não podem ser adicionadas aos sub-sites.



Figura 19 – Estrutura de um portal em SharePoint

5.4. Conteúdos Básicos do SharePoint

5.4.1. Tipos de Sites

Existem vários modelos de sites predefinidos que, na sua maioria, já se encontram equipados com algumas bibliotecas e listas específicas para cada tipo de site, cujas principais características encontram-se descritas a seguir.

No grupo de sites de colaboração destacam-se:

Site de equipa, para equipas de trabalho se organizarem, criarem e partilharem informações de forma rápida fornecendo uma biblioteca de documentos e listas para gerir anúncios, itens de calendário, tarefas e debates.

Site Wiki, onde uma comunidade pode debater e partilhar ideias. Fornece páginas Web que podem ser rapidamente editadas para registarem informações e, podendo ser interligadas através de palavras-chave.

No grupo de sites de empresa destacam-se:

Centro de registos, destinado à gestão de registos. Os gestores de registos podem configurar a tabela de encaminhamento para enviar ficheiros recebidos para localizações específicas, bibliotecas de documentos específicas.

Centro de procuras com separadores, inclui uma caixa de procura com dois separadores, sendo um para procuras gerais e outro para procuras de informações sobre pessoas. É possível adicionar e personalizar separadores conseguindo concentrar outros âmbitos de procura ou tipos de resultados.

No grupo de sites de publicação destaca-se:

Site de publicação com fluxo de trabalho, para publicar páginas Web numa agenda utilizando fluxos de trabalho de aprovação. Inclui bibliotecas de documentos e imagens para armazenar itens de publicação. Por predefinição, apenas os sites que têm este modelo podem ser criados neste site.

Grupo de sites personalizados:

No grupo de sites personalizados encontram-se os sites guardados como modelo, isto é, sites baseados num modelo construído pelo utilizador. Estes sites servem para rapidamente recriar o design, a estrutura e os conteúdos de um site já construído previamente, ajudando, desta forma, a manter a coerência entre os sites e minimizando o tempo de desenvolvimento.

Versão Móvel de Sites

O SP permite o desenvolvimento de sites para dispositivos móveis, como por exemplo um PDA. O endereço da versão móvel, *mobile*, do portal é o mesmo corresponde ao endereço do site seguido de "/m". Na figura 20 ilustra a versão móvel de um site SP num PDA.



Figura 20 – Versão *mobile* de um site SharePoint num PDA

5.4.2. Tipos de Conteúdo

Todos os componentes no SP têm por base os Tipos de Conteúdo, Content Types, CT, que corresponde a um grupo reutilizável de definições para uma categoria de conteúdo.

Os tipos de conteúdo são definidos ao nível do site e utilizados em listas e bibliotecas permitindo organizar, gerir e processar conteúdo de uma forma consistente ao longo de uma colecção de sites. Ao definir tipos de conteúdo para géneros específicos de documentos ou produtos de informação, uma organização pode garantir que cada um destes grupos de conteúdo é gerido de uma forma consistente.

Os CT servem também para especificar modelos e processos de um fluxo de trabalho. Por exemplo, um departamento de uma organização pode ter modelos para relatórios de actividades, documentos padrão, orçamentos e apresentações e, quando é criado um novo ficheiro a partir da biblioteca de documentos, é possível seleccionar qual o modelo a utilizar, cuja definição é feita através de CT.

Os tipos de conteúdo podem ser definidos para qualquer tipo de item no SP, incluindo documentos, itens de lista ou pastas. Cada CT pode especificar as colunas que se pretendem atribuir a itens deste tipo, o modelo de documento no qual basear novos itens deste tipo e os fluxos de trabalho disponíveis para itens de um tipo de conteúdo.

5.4.3. Listas

Uma lista é em tudo idêntica a uma tabela de uma base de dados, onde as colunas de uma lista correspondem às colunas da tabela, com a grande vantagem de poder armazenar vários tipos de dados, desde datas, documentos, hiperligações, imagens ou mesmo cálculos com base no valor de outras colunas bastando para isso escolher o tipo de coluna e a sua configuração. Por exemplo, uma coluna do tipo *Rich Text* disponibiliza automaticamente uma caixa de texto com uma barra de ferramentas que permite formatar o texto, inserir hiperligações, adicionar tabelas e imagens, etc, tudo isto feito num browser sem recurso qualquer programação.

Quando se cria um site em SP, são criados vários tipos de listas predefinidas que variam desde uma área de debate a uma lista de calendário. No entanto, é possível personalizar as listas predefinidas de várias formas ou criar listas personalizadas com as colunas pretendidas.

As listas podem beneficiar das funcionalidades de correio electrónico, se a recepção ou envio de correio estiver activado no site, e algumas listas, como calendários, anúncios, blogs e áreas de debate, podem ser configuradas para que as pessoas possam adicionar conteúdo enviando mensagens de correio electrónico. Outras listas, como as de tarefas e registo de problemas podem ser configuradas para enviar mensagens de correio electrónico a pessoas quando os itens lhes são atribuídos.

A figura 21 apresenta a interface para o utilizador onde são listadas todas as listas existentes no portal.











Listas		
	Anotação	3 Há 3 semanas
	Anúncios	Utilize a lista Anúncios para publicar mensagens na home page do site. 1 Há 4 semanas
	Calendário	Utilize a lista Calendário para se manter informado sobre reuniões, prazos e outros eventos importantes. 0 Há 4 semanas
	Conteúdo Reutilizável	Os itens desta lista contêm HTML ou conteúdo de texto que pode ser inserido em páginas Web. Se um item tiver a actualização automática seleccionada, o conteúdo será inserido em páginas Web como uma referência só de leitura e o conteúdo será actualizado se o item for alterado. Se o item não tiver a actualização automática seleccionada, o conteúdo será inserido como cópia na página Web e o conteúdo não será actualizado se o item for alterado. 0 Há 4 semanas
	Contrato	3 Há 4 dias
	Email	3 Há 3 semanas
	Equipamento	3 Há 3 semanas
	Evento	3 Há 3 semanas
	Faqs	3 Há 2 semanas
	Hiperligações	Utilizar a lista Hiperligações para obter hiperligações para páginas Web que os membros da sua equipa poderão considerar interessantes ou úteis. 0 Há 4 semanas

Figura 21 – Listas do SharePoint

5.4.4. Bibliotecas

Uma biblioteca é uma localização onde se pode criar, recolher, actualizar e gerir ficheiros. Cada biblioteca apresenta uma lista de ficheiros e informações sobre estes, o que ajuda as pessoas que utilizam os ficheiros e trabalhar em equipa. Deste modo, é possível criar e gerir documentos, folhas de cálculo, páginas Web, apresentações, formulários e outros tipos de ficheiros numa biblioteca.

É possível personalizar bibliotecas de várias formas, desde a forma como os documentos são visualizados, registados, geridos e criados, até ao controlo de versões, incluindo o número e o tipo da versão, principal ou rascunho, permitindo também limitar o que as pessoas podem ver antes dos documentos serem aprovados. A biblioteca de documentos partilhados é criada quando o SP cria um novo site.

Dependendo do tipo de ficheiros que se pretende armazenar é possível escolher o tipo de biblioteca mais adequado tendo em conta também a forma como se pretende utilizá-los, estando para isso disponíveis vários tipos de bibliotecas, as quais se encontram descritas a seguir.

A interface que permite ver as bibliotecas existentes é igual à apresentada para as listas, tal como se pode ver na figura 22.










Bibliotecas de documentos			
	Biblioteca de Estilos	Esta lista de sistema foi criada pela funcionalidade Recursos de Publicação para armazenar folhas de estilos de XSL e folhas de estilos em cascata personalizadas.	73 Há 4 semanas
	Doc. Contrato		6 Há 2 semanas
	Docs		1 Há 3 semanas
	Documentos	Esta biblioteca de sistema foi criada pela funcionalidade Publicação para armazenar documentos que são utilizados em páginas deste site.	0 Há 9 dias
	Documentos da Coleção de Sites	Esta biblioteca de sistema foi criada pela funcionalidade Recursos de Publicação para armazenar documentos que são utilizados em toda a coleção de sites.	0 Há 4 semanas
	Documentos Partilhados	Partilhe um documento com a equipa adicionando-o a esta biblioteca de documentos.	6 Há 2 semanas
	Imagens	Esta biblioteca de sistema foi criada pela funcionalidade Publicação para armazenar imagens que são utilizadas em páginas deste site.	1 Há 9 dias
	Imagens da Coleção de Sites	Esta biblioteca de sistema foi criada pela funcionalidade Recursos de Publicação para armazenar imagens que são utilizadas em toda a coleção de sites.	5 Há 2 semanas
	Páginas	Esta biblioteca de sistema foi criada pela funcionalidade Publicação para armazenar páginas que são criadas neste site.	2 Há 3 dias

Figura 22 – Bibliotecas do SharePoint

Biblioteca de documentos

Para diversos tipos de ficheiro, incluindo documentos, folhas de cálculo, ficheiros PDF, etc.

Biblioteca de imagens

Para partilhar um conjunto de imagens ou gráficos digitais. Apesar das imagens poderem ser armazenadas noutros tipos de biblioteca do SP, as bibliotecas de imagens têm várias vantagens, por exemplo, a partir de uma biblioteca de imagens é possível ver imagens numa apresentação de diapositivos, transferir imagens para o computador e editar imagens com programas gráficos compatíveis com o WSS.

Biblioteca de páginas Wiki

Para criar um conjunto de páginas Wiki interligadas. Uma Wiki permite que várias pessoas recolham informações de rotina num formato fácil de criar e modificar.

Biblioteca de formulários

Para armazenar um grupo de formulários baseados em XML. Uma biblioteca de formulários requer um editor XML ou um programa de estruturação XML que seja compatível com o WSS, tal como o Microsoft InfoPath.

Tal como referido no ponto 5.4.4, as bibliotecas oferecem inúmeras funcionalidades, das quais as mais relevantes se encontram descritas a seguir.

Aprovação necessária do documento

É possível especificar para que seja necessária a aprovação de um documento, para que este fique disponível para todos os utilizadores. Por conseguinte, os documentos permanecem num estado pendente até serem aprovados ou rejeitados por alguém que tenha permissão para tal. No entanto, é possível definir quais os grupos de utilizadores que podem ver um documento enquanto este se encontra no estado pendente. Neste caso, os fluxos de trabalho

podem ser utilizados para gerir consistentemente este tipo de processos, aprovação ou revisão de documentos.

Controlo de versões

O controlo de versões é bastante útil porque permite guardar um histórico com a evolução da elaboração dos documentos, os quais podem ser consultados a qualquer momento, como também, restaurar versões anteriores à actual.

Deste modo, quando o controlo de versões se encontra activo, é possível ver em que momento um item ou ficheiro foi alterado e quem o alterou. Do mesmo modo, é igualmente possível ver em que momento as propriedades ou as informações sobre o ficheiro foram alteradas. Por exemplo, se alguém alterar a data de conclusão de um item de lista, essa informação surgirá no histórico de versões.

No caso dos ficheiros, é igualmente possível ver comentários incluídos pelas pessoas sobre as respectivas alterações e consequentes aprovações ou rejeições. Para tal, é possível visualizar qualquer versão anterior sem substituir a versão actual. No caso das páginas Web .aspx, apenas é possível visualizar os detalhes sobre as alterações efectuadas às mesmas, e não as páginas efectivamente criadas.

No caso de ser necessário restaurar parte de um documento, por exemplo, por ter sido cometido um erro na versão actual, é possível substituir de forma fácil a versão actual por uma versão anterior, tornando-se, de seguida, a versão actual parte do histórico de versões.

Edição de ficheiros *offline*

É possível dar saída dos ficheiros, isto é, requisita-los e bloqueá-los, por forma a trabalhá-los em modo *offline*, utilizando para tal uma aplicação que seja compatível com o WSS, como por exemplo o Microsoft Office System 2007. No fim, deve-se dar entrada dos ficheiros alterados para que a respectiva versão de rascunho fique disponível para os outros utilizadores e, eventualmente, pendente para aprovação.

5.4.5. Vistas

A utilização de vistas para ver os itens de uma lista ou biblioteca permite visualizar de diferentes formas o mesmo conteúdo. Por exemplo, é possível criar vistas que apresentem os ficheiros mais recentes de uma biblioteca de documentos, ou dos itens de uma lista que se aplicam a um determinado departamento ou ainda dos ficheiros criados por uma pessoa em particular.

Para tornar as listas ou bibliotecas mais versáteis, é possível adicionar colunas adicionais e depois criar múltiplas vistas que mostram ou ocultam as colunas, dependendo do propósito da vista e dos dados mais significativos a serem apresentados em cada caso. Por exemplo, para uma mesma lista, um departamento pode estar interessado em ver apenas o título e a

descrição de uma tarefa, enquanto que outro departamento pode querer mais informações como a prioridade e o estado da mesma. As vistas podem também ajudar a gerir a forma como as listas e bibliotecas são visualizadas nos dispositivos móveis, dado que alguns dispositivos podem restringir o número de caracteres apresentados numa coluna.

Todas as listas e bibliotecas têm vistas predefinidas criadas com base no seu tipo e nas definições aplicadas. No entanto, algumas listas e bibliotecas têm outras vistas incorporadas específicas de cada uma, como por exemplo a vista de "calendário", específica da lista do tipo calendário, ou a vista de tarefas com "conclusão prevista para hoje", específica da lista do tipo tarefas.

Contudo, é possível criar todo o tipo de vistas personalizadas, com a conjugação de vários parâmetros, como por exemplo, filtragem, ordenação, agrupamento, paginação, estilos de apresentação, etc. A imagem 23 apresenta a interface para o utilizador escolher, via browser, um dos estilos definidos nos ficheiros da referida pasta de estilos.

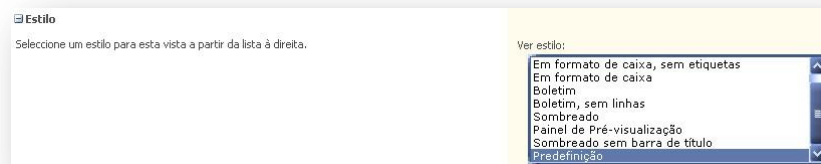


Figura 23 – Estilos de personalização de listas

Vistas Públicas e Pessoais

As vistas podem ser pessoais e públicas, sendo que uma vista pessoal só está disponível para um utilizador, enquanto que uma vista pública pode ser usada para construir a vista predefina, dos itens de uma lista ou biblioteca, para todos os utilizadores.

Tipos de Vistas

Quando se cria uma vista personalizada, é possível baseá-la num tipo de vista, que determina a forma como os itens de lista são apresentados.

Vista Padrão

Esta vista, que é a predefinição da maioria dos tipos de listas e biblioteca, permite apresentar os mesmos itens de uma lista sob a forma de uma lista linear, semelhante a uma tabela.

Vista de Calendário

A vista de calendário permite obter uma visão mais abrangente assim como também permite detalhar no nível de informação, podendo por isso ser mensal, semanal ou diária. Na figura 24 pode-se ver uma vista semanal seleccionada num calendário (1) com os eventos localizados de acordo com a data e hora (2).

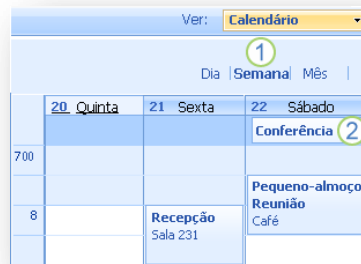


Figura 24 – Vistas de calendário do SharePoint

Vista de Folha de Dados

A vista de folha de dados fornece dados num formato que editável, tal como na folha de cálculo do Excel. Esta vista pode ser útil para efectuar grandes tarefas de edição ou personalização, ou exportar os dados para um programa de folha de cálculo ou base de dados.

Vista de Gantt

Quanto à vista de Gantt, esta fornece uma apresentação visual em barras horizontais que registam o progresso, caso os dados sejam baseados num intervalo de tempo. Uma vista de Gantt pode ajudar a gerir projectos e a ver uma descrição geral rápida dos dados. É possível utilizar esta vista para visualizar o progresso geral e, por exemplo, para ver quais as tarefas que se sobrepõem. Dado que esta vista não permite definir precedências, esta funcionalidade deve ser encarada como uma ferramenta que permite acompanhar a evolução de tarefas e/ou eventos de forma visual mas não como um típico gestor de projectos. Tal como se pode observar na figura 25, a vista de Gantt é bastante semelhante àquela apresentada pelo Microsoft Project.

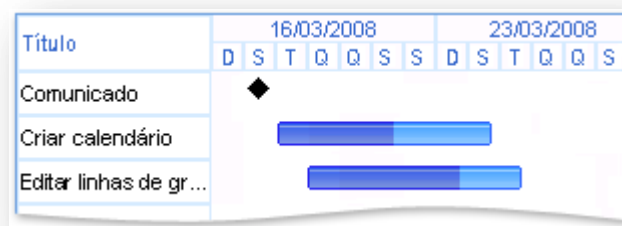


Figura 25 – Vista de Gantt do SharePoint

Existem ainda outros tipos de vistas, como por exemplo, no caso das bibliotecas, a vista de explorador, idêntica ao explorador de ficheiros do Windows.

Resumindo, as vistas fornecem várias opções para tornar as listas e bibliotecas mais eficazes, das quais se destacam algumas das formas de utilizar as vistas:

- Filtrar por um conjunto de critérios, como o nome da pessoa ou departamento.
- Ordenar, por exemplo, para mostrar os ficheiros modificados mais recentemente.
- Ocultar ou mostrar colunas, por exemplo, para se obter uma vista mais focada numa determinada informação.
- Agrupar informação baseada nos dados da lista, como por exemplo, agrupar por departamento.
- Mostrar subtotais das colunas, como por exemplo, o número de total documentos do tipo contrato.
- Ver uma vista de calendário de uma lista que tenha datas de início e datas de fim.
- Ver uma lista num programa de base de dados que permita a análise de dados.
- - Ver todos os itens ao mesmo nível numa vista plana sem pastas.

Mais uma vez, este tipo de requisitos é facilmente implementado sem recurso a qualquer tipo de programação.

5.4.6. Páginas

O SP apresenta vários modelos de páginas, a partir dos quais se podem criar novas páginas pelo browser. Esses modelos diferem apenas no esquema e organização da informação passível de ser preenchida numa página, bem como das Zonas de Peças Web, podendo estas últimas variar também em número.

A Zona de Peças Web são locais onde se podem adicionar Peças Web, cuja descrição se encontra no Anexo A. A figura 26 ilustra um exemplo de um modelo de página, onde se pode observar uma zona para inserir a imagem da página, uma zona para o conteúdo da página, duas zonas para hiperligações e três Zonas de Peças Web.

5.4.7. Peças Web

No SP, as páginas são constituídas por uma ou mais Peças Web, *Web Parts*, sendo esta a unidade de informação modular que serve de base para a construção visual das referidas páginas. As Peças Web podem ser inseridas em zonas de Peças Web existentes numa página e, em seguida, personalizadas de modo a criar uma página exclusiva para os utilizadores do site. Na figura 26 pode-se ver as várias zonas editáveis e de peças Web de uma página de SP, em modo editável.



Figura 26 – Página de SharePoint em modo editável

Existem várias Peças Web incluídas por predefinição em qualquer site de SP, que podem ser usadas e interligadas para criar uma vasta gama de soluções exclusivas.

A seguir são apresentadas algumas dessas Peças Web.

Peça Web Editor de Conteúdo

Esta Peça Web permite adicionar texto formatado, tabelas, hiperligações e imagens a uma página, oferecendo para tal um editor HTML do tipo WYSIWYG.

Peça Web Formulário

A peça Web formulário permite ligar e filtrar uma coluna de dados existente noutra Peça Web.

Peça Web Vista de Lista

As vistas de listas, descritas no ponto 5.4.5, acabam por ser Peças Web quando inseridas numa página, apresentando informações de várias formas com objectivos diferentes, como por exemplo, filtrar, ordenar ou seleccionar colunas específicas.

Peça Web Visualizador de Páginas

A Peça Web Visualizador de Páginas apresenta qualquer página Web, ficheiro ou pasta, ocupando apenas o espaço da respectiva Peça Web.

Peça Web Utilizadores do Site

A apresentação de uma lista de utilizadores e grupos que têm permissão para utilizar um site pode ser feita com recurso à Peça Web Utilizadores do Site.

O exemplo que se segue utiliza a Peça Web Imagem para descrever as funcionalidades básicas de uma Peça Web.



Figura 27 – Peça Web imagem

1 - Barra de título da Peça Web que contém o título da Peça Web.

2 - Menu da Peça Web que contém funções tais como minimizar, fechar ou editar a Peça Web ou ainda obter ajuda para uma Peça Web específica. Quando a página está no modo de edição, é igualmente possível utilizar este menu para eliminar a Peça Web ou ligá-la a outras Peças Web, dependendo do tipo de peça que estiver a ser utilizada. É, de facto, frequente encontrar páginas de SP com Peças Web ligadas entre si para, deste modo, interagirem e partilharem dados, apresentando conteúdos de uma forma dinâmica.

3 - Corpo da Peça Web que contém o conteúdo que tiver sido especificado para o tipo de Peça Web que se está a utilizar. Neste exemplo, esta peça é uma Peça Web Imagem, a qual apresenta uma imagem.

Uma página de Peças Web é um tipo especial de página Web na qual é possível utilizar Peças Web para consolidar dados, como, por exemplo, listas, gráficos, e conteúdo Web, como, por exemplo, texto e imagens.

A página de entrada de um site é um exemplo de uma página de Peças Web. Deste modo, é possível criar uma página de Peças Web seleccionando um dos modelos de site ou de páginas disponíveis ou, em alternativa, utilizando um programa de criação de Web compatível com o WSS, como, por exemplo, o Microsoft Office SharePoint Designer 2007, para criar uma página de Peças Web de raiz.

Propriedades das Peças Web

Cada Peça Web partilha um conjunto de propriedades comuns, também denominadas propriedades de classe de base, organizadas em secções no painel de ferramentas que controlam o aspecto da Peça Web, como por exemplo, o título, altura e largura, o esquema desta, como por exemplo, a ordem das Peças Web na zona e a direcção do conteúdo, e

características avançadas da mesma, como por exemplo, o ícone e descrição da imagem, tal como se pode ver na figura 28.

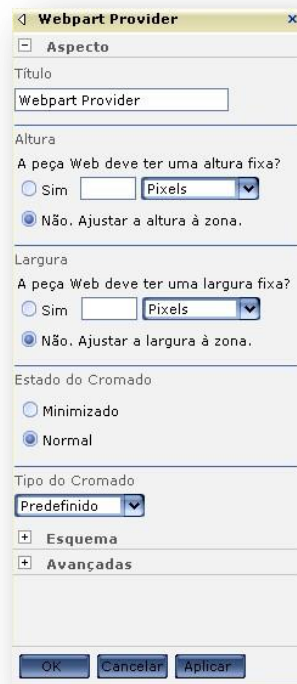


Figura 28 – Menu de personalização de uma Peça Web

Muitas Peças Web possuem igualmente propriedades personalizadas exclusivas desta. Estas propriedades são habitualmente apresentadas sobre ou sob as propriedades comuns das Peças Web no painel de ferramentas. Por exemplo, a Peça Web Imagem possui propriedades personalizadas adicionais, incluindo a hiperligação de imagem, o respectivo alinhamento horizontal e vertical e a cor de fundo.

Vistas de Peça Web

É também possível personalizar uma Peça Web com dois tipos de vistas, a partilhada, cujas alterações efectuadas se reflectem para todos os utilizadores, e a pessoal, onde as alterações estarão apenas disponíveis para o próprio utilizador. De referir que, para alterar uma vista partilhada é preciso ter permissões para tal.

Peças Web e ligações de peças Web

Outra funcionalidade das Peças Web é a possibilidade de as ligar facilmente, tal como já foi referido acima 5.4.7, permitindo que estas partilhem dados entre si de modo a sincronizar o respectivo comportamento. Ao ligar Peças Web, é possível, por exemplo, apresentar dados em

vistas alternadas, efectuar cálculos relacionados entre duas Peças Web e filtrar uma Peça Web utilizando valores de outra, tudo isto na mesma página.

5.4.8. Audiências

Ao utilizar audiências alvo, é possível apresentar e direccionar conteúdos como itens de lista ou de biblioteca, hiperligações de navegação e Peças Web para grupos específicos de pessoas. Isto é útil quando se pretende apresentar informações diferentes a grupos distintos de pessoas, apresentando desta forma apenas a informação mais relevante para cada utilizador. A título de exemplo, é possível adicionar uma Peça Web à página de um departamento que contém uma lista de contratos visíveis apenas para esse departamento.

5.4.9. Fluxos de Trabalho

Os fluxos de trabalho ajudam as pessoas a colaborar com documentos e a gerir tarefas de projecto através da implementação de processos em documentos e itens de bibliotecas e listas, num site do MOSS 2007. Os fluxos de trabalho ajudam também as organizações a aderirem a processos consistentes, melhorando desta forma a eficiência e produtividade da organização, através da gestão de tarefas e respectivos passos envolvidos nos processos. Isto permite que as pessoas que desempenham essas tarefas se concentrem na realização do trabalho em vez de se ocuparem da gestão do fluxo de trabalho.

Um site do MOSS 2007 oferece vários fluxos de trabalho predefinidos que visam cenários comuns, como por exemplo:

Aprovação

O fluxo de trabalho de Aprovação encaminha um documento ou item para um grupo de pessoas o aprovar. Por predefinição, este fluxo de trabalho é associado ao tipo de conteúdo Documento, pelo que fica automaticamente disponível em bibliotecas de documentos. Uma versão do fluxo de trabalho Aprovação também é associada por predefinição à biblioteca Páginas de um site de publicação e pode ser utilizada para gerir o processo de aprovação de páginas Web com vista à sua publicação.

Recolher Comentários

Este fluxo de trabalho encaminha um documento ou item para um grupo de pessoas o comentar. Os revisores podem enviar comentários que são compilados e enviados para a pessoa que gerou o fluxo de trabalho. Por predefinição, o fluxo de trabalho de Recolha de

Comentários é associado ao tipo de conteúdo do Documento, pelo que fica automaticamente disponível em bibliotecas de documentos.

Aprovação de Eliminação

Este fluxo de trabalho, que suporta processos de gestão de registos, gere a retenção e expiração de documentos, permitindo que os participantes decidam se pretendem reter ou eliminar os documentos cujo prazo expirou. O fluxo de trabalho de Aprovação de Eliminação destina-se a ser utilizado sobretudo num site do Centro de Registos.

Três estados

O fluxo de trabalho de Três Estados permite definir diferentes acções conforme o estado em que se encontre, podendo ser utilizado para gerir processos que requeiram um rastreio de um grande volume de assuntos ou itens, como questões de suporte a clientes, potenciais clientes ou tarefas de projecto.

Suporte para fluxos de trabalho personalizados

Embora os fluxos de trabalho predefinidos disponíveis no MOSS 2007 possam ser personalizados de modo a dar resposta a diferentes necessidades, é possível optar por conceber e estruturar fluxos de trabalho exclusivos, com auxílio de outras ferramentas como o Microsoft Office SharePoint Designer 2007, onde os fluxos de trabalho são criados a partir de uma lista de actividades sem necessidade de programação, ou o Visual Studio, que permite a inclusão de código personalizado.

5.4.10. Reciclagem

A reciclagem do MOSS 2007 permite recuperar itens eliminados e encontra-se organizada em duas vertentes, sendo a primeira a nível local e a segunda a nível da colecção de sites.

Por predefinição, os itens na reciclagem do primeiro nível expiram ao fim de duas semanas, passando para a reciclagem do segundo nível, onde são completamente eliminados passados três meses. No entanto, é possível definir políticas de gestão para a reciclagem onde, entre outras coisas, se pode alterar estes prazos.

Uma das mais poderosas funcionalidades da reciclagem é o histórico de acções efectuadas sobre um item, que permite saber todas as acções tomadas sobre este, mesmo que já não exista.

5.4.11. Permissões

O MOSS 2007 fornece alguns grupos de utilizadores predefinidos, aquando da criação de um site, com vários níveis de permissão igualmente predefinidos. É, no entanto, possível criar novos grupos com outras permissões, permitindo desta forma que se configure adequadamente as permissões de todos os utilizadores do portal. A seguir encontram-se descritas as diferentes permissões que o SP oferece.

Controlo Total

Este nível de permissão contém o máximo de permissões, sendo normalmente atribuído ao grupo de Proprietários do site, por predefinição. Este nível de permissão não pode ser personalizado nem eliminado.

Estruturar

Permite que os utilizadores deste grupo criem listas e bibliotecas de documentos, editem páginas e apliquem temas, limites, folhas de estilo ou outros elementos de personalização no site. Os utilizadores com esta permissão podem ainda visualizar, adicionar, actualizar, eliminar e aprovar itens ou documentos de listas ou bibliotecas.

Gerir Hierarquia

Os utilizadores com estas permissões podem criar sites e editar páginas, itens e documentos de listas ou bibliotecas.

Aprovar

Com este nível de permissão é apenas possível editar e aprovar páginas, itens e documentos.

Contribuir

Este nível de permissão permite visualizar, adicionar, editar e eliminar itens e documentos de listas e bibliotecas existentes, sendo normalmente atribuído ao grupo de Membros do site, por predefinição.

Ler

Permite acesso só de leitura ao site, ou seja, os utilizadores e grupos com este nível de permissão podem ver páginas e abrir itens ou documentos. Por predefinição é atribuído ao grupo de Visitantes do site.

Acesso Limitado

Com este nível é possível visualizar páginas e documentos, mas não é possível ver versões históricas ou rever informações de direitos de utilizador. O nível de permissão Acesso

Limitado foi concebido para ser combinado com permissões detalhadas para conceder aos utilizadores acesso a uma determinada lista, biblioteca de documentos, item ou documento, sem lhes conceder acesso a todo o site. No entanto, para aceder a uma lista ou biblioteca, por exemplo, um utilizador tem de ter permissão para abrir o site principal e ler os dados partilhados, como o tema e barras de navegação no site. O nível de permissão Acesso Limitado não pode ser personalizado nem eliminado.

Estas permissões não são exclusivas, pelo que, é possível seleccionar várias destas permissões para o mesmo grupo de utilizadores. Caso uma aplicação desenvolvida em MOSS necessite de uma gestão de permissões muito específica, é sempre possível personalizar um grupo existente ou mesmo criar um novo grupo de utilizadores para incluir apenas os níveis de permissão pretendidos.

5.4.12. Hierarquia e herança

A figura 29 apresenta um resumo dos três tipos de permissões e segurança do SharePoint, que são os níveis apresentados no ponto anterior, os utilizadores e grupos que se podem criar e os objectos aos quais se pode atribuir permissões específicas. Por exemplo, é possível definir num documento permissões de acesso exclusivas a um utilizador ou grupo de utilizadores.



Figura 29 – Tipos de permissões do SharePoint

Por predefinição, as permissões nas listas, bibliotecas, pastas de listas e bibliotecas, itens e documentos são herdadas do respectivo site principal. No entanto, também é possível interromper esta herança para qualquer objecto protegido num nível inferior da hierarquia, ou seja, editando as permissões desse objecto projecto e criando uma atribuição de permissão única.



Figura 30 – Herança de permissões

Na figura 30, o sub-site 1 herda permissões do site de nível superior, o que significa que as alterações efectuadas aos grupos de SP e níveis de permissão no site de nível superior também afectam o sub-site 1.

O sub-site 2 também está a herdar permissões do site hierarquicamente acima, neste caso o sub-site 1. No entanto, como o sub-site 1 está a herdar permissões do site principal, as alterações efectuadas aos grupos e níveis de permissão no site de nível superior afectam também o sub-site 2. Deste modo, a gestão de permissões nos sub-sites 1 e 2 é feita no site de nível superior, sendo, no entanto, possível interromper esta hierarquia de permissões a qualquer momento. Em relação ao sub-site 3, este tem permissões exclusivas, pelo que, qualquer alteração nas permissões do site de nível superior não afectam as permissões deste. Neste caso, o sub-site 4 herda as permissões do sub-site 3.

5.4.13. Pesquisa

A pesquisa do MOSS 2007 permite obter resultados de documentos, pessoas, informações, páginas e outros conteúdos, apresentando os resultados por relevância e respeitando as políticas de segurança e as permissões de cada utilizador.

A pesquisa tem alguns mecanismos que melhoram o seu desempenho e usabilidade, como por exemplo, a indexação dos dados, para consultas rápidas e eficazes, e a oferta de sugestões e sinónimos, para alertar eventuais enganos do utilizador.

Os resultados da pesquisa são apresentados de uma forma clara, evidenciando as palavras pesquisadas, o tipo de conteúdo através de ícones, mostrando a hiperligação, a data de modificação, entre outros aspectos. Tal como se pode observar na figura 31, oferece também funcionalidades tais como paginação, ordenação de resultados, subscrição de alertas e RSS. Por predefinição, o serviço de pesquisa encontra-se desactivo no WSS 3.0, sendo necessário configurá-lo no Centro de Administração, cujos detalhes se encontram no Anexo A.



Figura 31 – Resultados da pesquisa do SharePoint

5.4.14. Integração com o Microsoft Office 2007

O MOSS 2007 tem uma forte integração com as aplicações do Microsoft Office System 2007, das quais se destacam o Word, Excel, PowerPoint, Outlook, Project, OneNote e InfoPath. Deste modo, é possível interagir com informação armazenada num portal SharePoint sem ser necessário transferir os conteúdos manualmente.

No MOSS 2007 é possível criar bibliotecas de documentos baseadas em modelos de documentos do Microsoft Office System 2007, como por exemplo um modelo do Word 2007, previamente construídos para o efeito, aos quais se pode acrescentar meta-dados associados a cada documento.

É possível guardar folhas de cálculo do Excel 2007 num site de SP, permitindo assim que outros utilizadores possam aceder às mesmas através de um browser, sem contudo terem acesso a informações proprietárias, como por exemplo fórmulas e modelos financeiros incorporados na folha de cálculo.

Com o Outlook 2007 é possível estabelecer uma ligação directa com uma lista, que corresponde a uma pasta na aplicação de e-mail, criar uma conta de SharePoint no Outlook e configurar algumas opções, como por exemplo, a sincronização automática ou manual.

Entre outras funcionalidades, é possível partilhar e-mails, contactos, calendários e tarefas entre o SharePoint e o Outlook. De facto, através da configuração da recepção de e-mails, o MOSS 2007 permite aos utilizadores trocar informações, mensagens e ficheiros directamente entre diferentes áreas do portal, facilitando desta forma a comunicação e trazendo como principal benefício o armazenamento centralizado de e-mails, com os respectivos anexos, e consequente possibilidade de pesquisa de informação. No caso da partilha de tarefas, o Outlook pode receber documentos para serem trabalhados em modo offline e posteriormente actualizados no servidor quando ligado ao SharePoint.

O OneNote 2007 permite a criação de blocos de notas partilhados e armazenados no SP, permitindo desta forma que várias pessoas trabalhem sobre o mesmo bloco, sincronizando e coordenando informação.

É também possível desenvolver formulários no InfoPath com base em listas de SP, sendo necessário, para tal, a criação de uma ligação ao servidor, de modo a ser possível aceder aos campos de listas e bibliotecas do site para o qual se está a construir o formulário.

Quando integrado com o Office Project Server 2007, que inclui a aplicação Microsoft Project 2007, é possível partilhar áreas de trabalho de projectos, o que permite estender as tarefas, os documentos e os problemas do projecto pelos intervenientes do mesmo. Com esta integração, fica disponível no WSS um novo tipo de lista que estende as funcionalidades da já conhecida lista de Tarefas, como por exemplo as dependências num diagrama de Gantt.

Com o Access 2007 é possível manter uma cópia local de listas do SP em modo *offline*, para consulta, edição e posterior sincronização. As funcionalidades de relatório do Access permitem ainda analisar dados e criar relatórios, como se fosse qualquer outra tabela no Office Access 2007.

Na figura 32 pode-se observar alguns exemplos de integração do SP com o Microsoft Office System 2007, onde estão destacados a meta-informação de uma biblioteca num documento Word, a utilização desses mesmos campos na construção de um formulário InfoPath, a criação de um novo bloco de notas do OneNote partilhado num servidor de SP e a integração com o Outlook, Excel e Access.

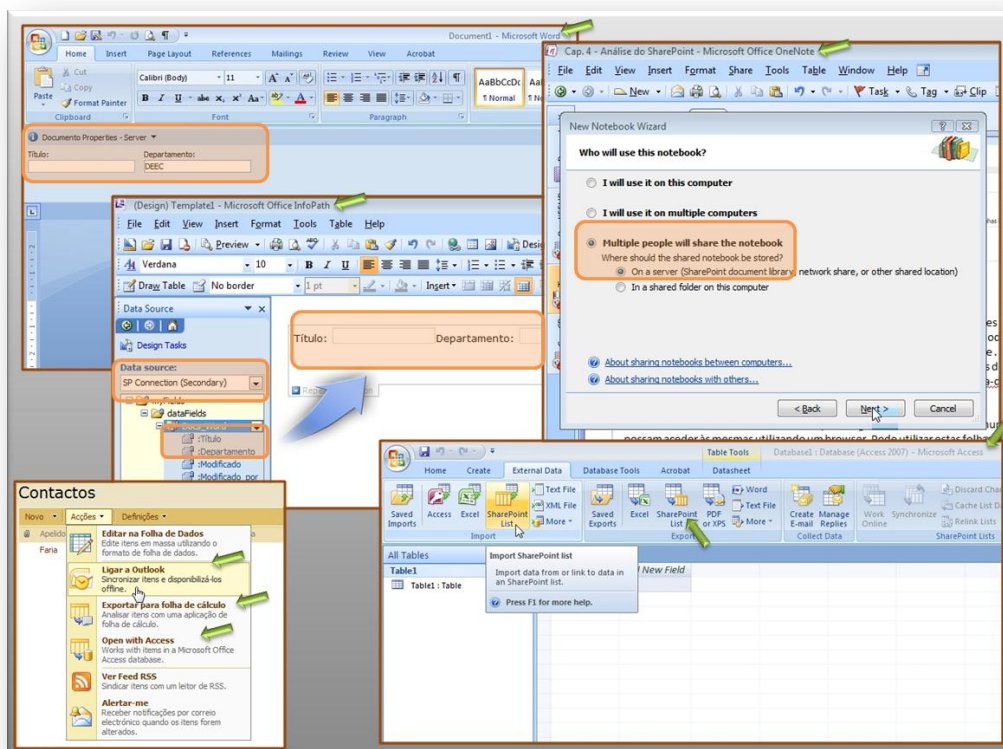


Figura 32 – Integração do SharePoint com o Office

5.4.15. Business Intelligence

O Excel Services, parte integrante do *Business Intelligence* do MOSS 2007, estende as capacidades do Excel 2007 através da partilha de folhas de cálculo, aumento da flexibilidade e segurança, apresentação de uma interface interactiva baseada na Web e facilidade de reutilização de modelos de folhas de cálculo, usando, para tal, o serviço de cálculo do servidor.

É possível disponibilizar toda esta informação através de *dashboards*, centros de registos, indicadores chave de desempenho (KPI), etc.

5.5. Funções Avançadas do SharePoint

No anexo A encontra-se uma descrição mais detalhada sobre a utilização do Centro de Administração, do SharePoint Designer como ferramenta de personalização e do Visual Studio como ferramenta de desenvolvimento de código.

Relativamente ao SharePoint Designer, são abordados temas como:

- Página Mestra
- Esquema de Página
- Personalização de listas e formulários de listas
- Fluxos de trabalho
- Ligações de Peças Web
- Personalização de ícones

Em relação ao desenvolvimento de código, são abordados temas como:

- Desenvolvimento de uma Peça Web
- Acesso programático a uma lista
- Utilização de AJAX
- Ligações de Peças Web
- Ligações de Peças Web com AJAX
- *Event Handlers*
- CAML
- LINQ para SharePoint
- SmartPart

5.6. Síntese

Não há dúvidas que o WSS ajuda organizações, empresas, equipas e unidades de negócio a serem mais eficientes através da facilidade de ligação de pessoas e partilha de informações. É uma infra-estrutura para a colaboração e uma base para a criação de aplicações Web versáteis e produtivas sem qualquer recurso a programação específica, embora isso também seja uma possibilidade para soluções mais personalizadas.

O MOSS 2007 oferece uma vasta gama de funcionalidades que cobrem áreas como a Colaboração, Portal Web, Pesquisa, Gestão de Conteúdos, Formulários de Processos de Negócio e *Business Intelligence*.

A Gestão Documental, a Gestão de Utilizadores, a produção de conteúdos Web por pessoas não especializadas, os sistemas de fluxos de trabalho, a colaboração, o suporte para a publicação multilingue, a pesquisa rápida e eficaz que este oferecem e a sua forte integração com programas e servidores do Microsoft Office System 2007, entre muitas outras temáticas, fazem do MOSS 2007 uma ferramenta completa capaz de dar resposta às necessidades das organizações e empresas, na procura da melhoria de produtividade e organização dos seus processos.

Na perspectiva de decisão entre a utilização do WSS ou do MOSS, uma vez que o MOSS implica a obtenção de uma licença, enquanto que o WSS é livre, a resposta é simples, pois tudo depende das funcionalidades que se pretendem implementar, com que esforço e com que tempo. Obviamente, o WSS oferece menos funcionalidades mas para uma dada aplicação podem ser suficientes.

Capítulo 6

Desenho e Implementação

Este capítulo descreve as três abordagens que foram testadas e avaliadas para a implementação do sistema de gestão de projectos no SP. A primeira abordagem é baseada exclusivamente no desenvolvimento de *User Controls*, a segunda com recurso ao ShrePoint Designer e a terceira através do desenvolvimento de código específico, dando assim liberdade a uma personalização total da aplicação. Estas três abordagens foram feitas na tentativa de desenvolvimento do sistema e exploração do SharePoint.

O facto de se ter considerado estas três abordagens resultou da necessidade de se desenvolver um primeiro protótipo e do facto dos conhecimentos serem limitados.

Este protótipo serviria para testar conceitos, avaliar a reacção dos utilizadores e descobrir novas áreas de interesse, sugestão dada até pelo próprio paradigma de State-Flow no ponto 3.5.

Por fim, este capítulo apresenta também o modelo desenhado para implementar o sistema e resume o que foi desenvolvido para essa implementação.

6.1. Abordagens

Dadas as circunstâncias referidas nos parágrafos acima, numa primeira fase começou-se por desenvolver apenas algumas funcionalidades básicas, como a marcação de tarefas, a anotação de eventos sobre a evolução do projecto e o armazenamento de documentos.

Para tal, foi utilizada uma biblioteca de documentos que o SP oferece e uma base de dados externa onde era guardada toda a informação relativa a projectos, fases, tarefas e anotações.

A utilização de uma base de dados externa em detrimento das listas de SP deveu-se ao facto de, na altura, já haver conhecimentos para o desenvolvimento de *User Controls*, e respectiva utilização da SmartPart, apresentada no Anexo A, e ainda não ter sido explorado o acesso programático aos elementos de SP.

A utilização apenas de listas de SP também não era uma solução viável na altura porque ainda não se possuía conhecimentos suficientes para, por exemplo, juntar na mesma página a listagem das anotações e o respectivo formulário de inserção, o que implicaria que o utilizador tinha que navegar para outra página para inserir uma nova anotação, situação essa que não era prática.

Esta abordagem implicaria a construção de duas páginas, para além da página inicial onde estavam listados os projectos em curso e concluídos, e as últimas anotações colocadas de todos os projectos. A página seguinte listava as fases do projecto, com base no identificador único, ID, recebido da página inicial. A última página deste sistema de navegação listava as últimas anotações e tarefas da fase seleccionada, mais uma vez com base no ID recebido.

Contudo, esta solução rapidamente se desvaneceu com a carência de relações entre os objectos da base de dados e os documentos da biblioteca do SP.

O passo seguinte passou por utilizar apenas objectos do SP e tentar personalizar com o SharePoint Designer. Neste esforço, descobriram-se as funcionalidades desta ferramenta, apresentadas no capítulo anterior, e conseguiu-se construir a maquete com razoáveis funcionalidades, algum dinamismo e agilidade.

Deste modo, foram criadas listas de tarefas e anotações e uma biblioteca de documentos, tudo isto com diferentes vistas de apresentação. Foram ainda criadas listas de contactos e hiperligações que poderiam ser úteis.

Nesta solução, conseguiu-se conjugar estes elementos na mesma página e dotar-lhes de alguns comportamentos, como por exemplo mostrar e ocultar a apresentação de todas as anotações, através de *JavaScript*.

Foi nesta fase que se sentiu a necessidade de explorar a linguagem XSLT, apresentada no Anexo A, para apresentar a lista de anotações, com as seis últimas anotações inseridas, ordenada de forma crescente. Sem esta personalização no SharePoint Designer, apenas era possível apresentar a lista com os seis primeiros itens ordenados de forma crescente ou os seis últimos itens ordenados de forma decrescente, situação descrita no Anexo A.

Na figura 33 pode-se ver o aspecto da página de um projecto, onde se encontram vários elementos na mesma página para permitir o acesso rápido aos mesmos.

Contudo, esta solução apresentava um grande problema porque cada projecto e cada fase não podiam ser itens de lista uma vez que, se fosse apresentada a lista de projectos numa página, a hiperligação intrínseca a cada item apontava para a página de detalhes do mesmo em vez de permitir a navegação para a página de fases desse projecto, pormenor que não podia ser configurado.

Isto implicava que cada projecto e cada fase fossem páginas independentes com as navegações mantidas manualmente, ou seja, se fosse necessário acrescentar uma nova fase seria preciso criar uma página nova, editá-la para colocar as Peças Web necessárias e depois

actualizar a página que listava as fases todas para acrescentar uma nova hiperligação para esta nova página, situação incomportável.

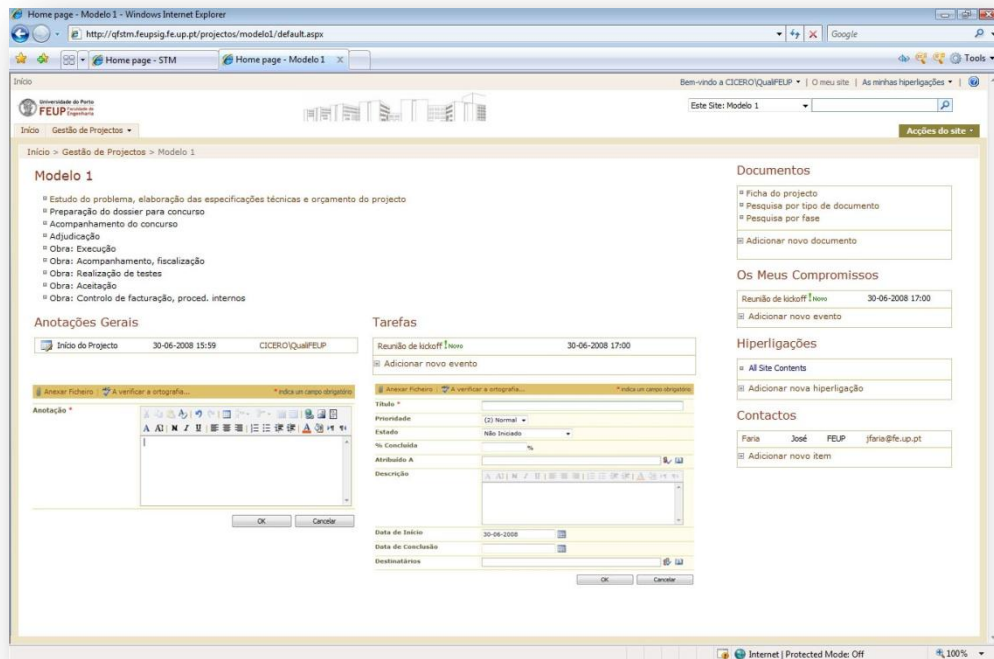


Figura 33 – Interface da segunda abordagem ao SharePoint

Quanto à biblioteca de documentos, inicialmente estava configurada como uma biblioteca normal com alguns campos adicionais para guardar informações sobre o tipo de documento e a fase à qual pertence. Contudo, isto implicaria a introdução de muita meta-informação aquando da adição de um novo documento, situação morosa e que não agilizava o processo.

Passou-se para a situação seguinte que correspondia a ter uma hiperligação em cada página de fase para adicionar um novo documento, sendo passado um parâmetro que identificasse a página onde o utilizador se encontrava anteriormente e, conseqüentemente, a respectiva fase.

Para tal, foram adicionados diferentes tipos de conteúdo, os *Content Types* referidos no capítulo 5, à biblioteca de documentos de modo a que, quando se adicionasse um novo documento, ficasse logo à fase onde o utilizador se encontrava no momento, sendo depois apenas necessário completar a informação através da escolha do tipo de documento que estava a ser inserido, por exemplo, contrato, orçamento, etc.

Para este funcionamento ser possível, era necessário copiar o URL da página de adição de um documento daquele tipo de conteúdo e criar uma hiperligação estática na página de fase correspondente.

Se fosse necessário criar uma nova fase, era necessário criar o tipo de conteúdo correspondente, adicioná-lo à biblioteca de documentos, copiar o URL da página de inserção de novo documento daquele tipo de conteúdo e, por fim, criar a hiperligação na nova página de

fase correspondente, situação absolutamente impensável quando um sistema deste tipo estiver a ser utilizado pelos gestores de projectos.

Para além disso, as soluções que o SharePoint Designer permitiam implementar não satisfaziam na totalidade o que se pretendia para a aplicação, ou quando satisfaziam, era necessário um esforço demasiado de tempo e trabalho para se conseguir o que se pretendia, o que levou ao abandono deste tipo de personalização.

Por fim, partindo do princípio que poderia ser tudo suportado por Peças Web personalizadas, desenvolvidas em VS, investiu-se na exploração deste tipo de programação e construiu-se um modelo da aplicação com os objectos de SP.

Começou-se então por se estabelecer quais os pontos que fariam parte da primeira versão da aplicação.

Resumidamente, a ferramenta estaria dotada com um espaço para marcação de qualquer tipo de evento, com um sistema de alertas apropriado, registo de anotações, troca de mensagens e armazenamento de documentos e e-mails. Para além disso, todos os factores que proporcionam uma boa experiência de utilização, descritos no capítulo 4, não seriam obviamente esquecidos, assim como o espaço com a vista integrada dos vários projectos.

6.2. Desenho

Para dar resposta a esta última abordagem, foram criados três conceitos, áreas de trabalho, módulos e componentes, onde as entidades definidas na tabela 2 se encaixam, tal como se pode ver na tabela 3.

Tabela 3 – Mapeamento de áreas de trabalho para objectos de SharePoint

Objecto em SP correspondente	
Áreas de Trabalho	
Projectos	Lista
Fases	Lista
Eventos	Lista
Documentos	Biblioteca de Documentos
Anotações	Lista
E-mails	Lista
Contactos	Lista
Problemas	Lista

Tabela 4 – Mapeamento de módulos para objectos de SharePoint

Objecto em SP correspondente	
Módulos	
Mensagens	Lista

Considera-se uma área de trabalho o espaço correspondente a uma entidade ou objecto principal que agrega outras áreas de trabalho, módulos e componentes de entidades ou objectos secundários, na sua maioria relacionados, tal como se pode ver na figura 34.

A disposição e localização de todos estes elementos presentes na interface poderia ser optimizada com base na experiência dos utilizadores.



Figura 34 – Interface do modelo a implementar

Quando se pretende navegar para outra área de trabalho, pode-se seleccionar a área pretendida, passando esta para principal e a outra para secundária, tal como se pode ver na figura 35.

Um módulo é um espaço que também corresponde a uma entidade ou objecto principal, agregando outros módulos, componentes e áreas de trabalho, embora não relacionados, pois tal relação não se justifica.

Qual a diferença principal entre um módulo e uma área de trabalho? Numa área de trabalho é possível, por exemplo, visualizar as fases de um projecto e, seleccionando uma fase em particular, são actualizadas todas as outras áreas de trabalho em conformidade com a fase seleccionada, isto é, documentos, anotações, eventos, e-mails, contactos e problemas da área seleccionada. Num módulo, não há essa sincronização entre o objecto principal do módulo e as áreas de trabalho presentes na interface.

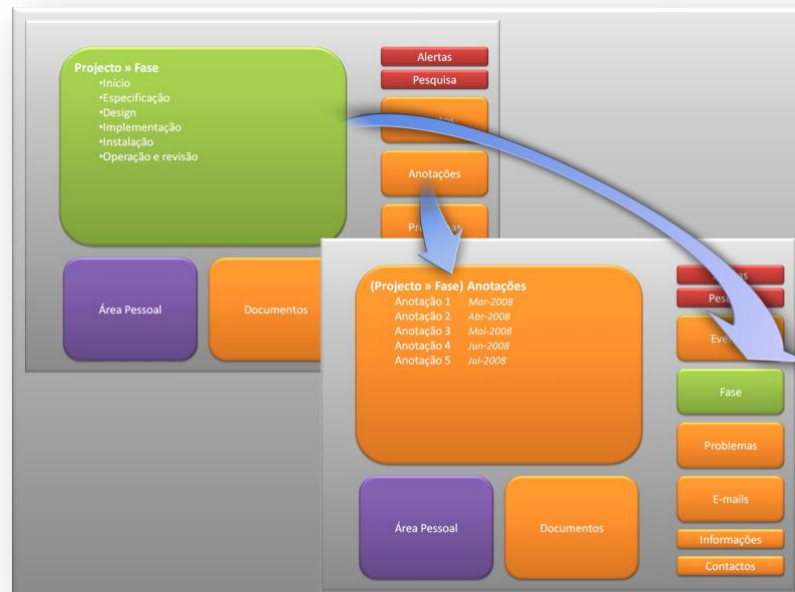


Figura 35 – Exemplo de navegação entre as áreas de trabalho do modelo

Os componentes correspondem a informações úteis que se encontram disponíveis na interface, como informações relacionadas com o projecto, uma caixa de pesquisa rápida, alertas de eventos actuais e um bloco pessoal para eventos e anotações pessoais.

O utilizador pode personalizar a interface da figura 34 conforme os seus gostos e preferências, sem com isso alterar a interface dos outros utilizadores. Esta personalização já é uma funcionalidade que o SharePoint oferece, não sendo por isso necessário qualquer desenvolvimento adicional.

Listas e Bibliotecas Criadas

As listas de projectos e fases permitem manter facilmente os respectivos projectos e fases, possibilitando a adição de novos projectos e a edição e eliminação dos mesmos, facilitando também a relação com outros objectos.

A lista de eventos é bastante completa e modular, permitindo a gestão de tarefas, compromissos profissionais e pessoais, ou outros eventos. Esta lista permite definir se o evento é para todo o dia ou se decorre apenas dentro de horas específicas, permite também definir a periodicidade do evento, bem como a prioridade e o estado, entre outras informações.

Desta lista é possível apresentar variadas vistas, em forma de tabela, de calendário ou de Gantt, sobre as actividades todas, quer profissionais quer pessoais, sendo esta uma vantagem de misturar toda esta informação na mesma lista. Assim, é possível apresentar, por exemplo, “As Minhas Tarefas”, “Tarefas do Projecto para Hoje”, “Os Meus Compromissos Pessoais”, “As Minhas Tarefas de Todos os Projectos” etc.

A biblioteca de documentos permite armazenar os documentos relativos ao projecto, classificando-os apenas quanto ao tipo de documento, por exemplo, se é um contrato, um orçamento, um relatório, etc. Desta biblioteca é também possível apresentar várias vistas, por tipo de documento, por fase a que pertence, etc, bem como a já referida vista de explorador (ponto 5.4.4).

A lista de anotações serve apenas de registo de evoluções, não passando por isso de uma caixa de texto.

A lista de e-mails serve para armazenar os e-mails directamente da aplicação de cliente de e-mail. A integração com esta aplicação de cliente de e-mail terá que ser feita com código para permitir uma gestão mais fácil dos e-mails na aplicação cliente, ou seja, por exemplo, para permitir guardar apenas parte do e-mail ou guardar os anexos na biblioteca de documentos.

A lista de contactos armazena os contactos relativos ao projecto.

A lista de problemas suporta a comunicação de eventuais problemas que tenham surgido.

De igual modo, a lista de mensagens suporta a comunicação entre utilizadores e serve também de armazenamento de mensagens pessoais, bastando para isso declarar essa intenção na opção do formulário. Se estiver seleccionada a opção de mensagem para outro utilizador, é apresentado um novo campo para se inserir os utilizadores a quem se destina a mensagem, caso contrário, este campo é ocultado. As mensagens pessoais funcionam como uma espécie de anotações pessoais, típicas de serem escritas em *Post Its*, com a vantagem de aqui ser possível definir uma data de alerta.

As apresentações dos dados destas listas (anotações, e-mails, contactos, problemas e mensagens) dependem do contexto em que se está a consultar, existindo por isso diferentes vistas também.

Para além destas listas criadas para a aplicação do modelo, existem outras listas que não estão aqui contempladas por já se encontrarem incluídas em qualquer site de SP, como por exemplo a lista de utilizadores.

Os alertas são chamadas de atenção que aparecem na interface sempre que haja um alerta. Por exemplo, se algum utilizador guardar uma mensagem para si e definir uma data de alerta, esta aparecerá sob a forma de alerta no dia especificado.

As informações reúnem dados do projecto, como por exemplo a ficha do projecto e a sua execução financeira, para além de apresentar indicadores estatísticos sobre a evolução do

projecto. Nesta fase da implementação, a ficha do projecto não passa de um ficheiro e, tanto a execução financeira como os indicadores estatísticos de tempo, custos e qualidade ficarão para futuras evoluções desta aplicação.

Para que todos estes objectos estejam interligados entre si, foi ainda criada uma lista adicional, lista relação, que mantém todas as relações entre objectos, isto é, armazena todos os identificadores únicos de cada objecto que estejam relacionados. Numa primeira fase, essa relação é feita par a par, tal como se pode ver na figura 36.

Relação							
ID Relação	ID Projecto	ID Fase	ID Evento	ID Documento	ID Anotação	ID E-mail	ID Contacto
1	2	1					
1	2	2					
1		2		5			
1				5	3		

Figura 36 – Lista de relação do modelo

Estas relações são automaticamente detectadas em função do local onde o utilizador se encontra. Por exemplo, se o utilizador se encontrar dentro de uma fase de um projecto e adicionar um novo documento, este fica automaticamente associado à fase. Contudo, no momento em que se adiciona esse novo objecto, há sempre a possibilidade de manifestar a intenção de não associar a nada ou, eventualmente, associar a outros objectos.

As listas e bibliotecas identificadas nas tabelas 3 e 4, fazem parte do seguinte modelo de dados, esquematizado na figura 37.

De referir que, para além das colunas apresentadas no modelo da figura 37, o SharePoint inclui sempre, por predefinição outras colunas, como são o caso das colunas "Criado", "Criado Por", "Modificado", "Modificado Por" e "Versão", entre outras menos relevantes para este modelo. Este modelo é semelhante a um modelo de uma base de dados relacional pois, de facto, com a utilização de soluções personalizadas é possível dotar o modelo de listas do SharePoint com restrições de integridade e actualização de dados em cascata, tal como acontece numa base de dados normal.

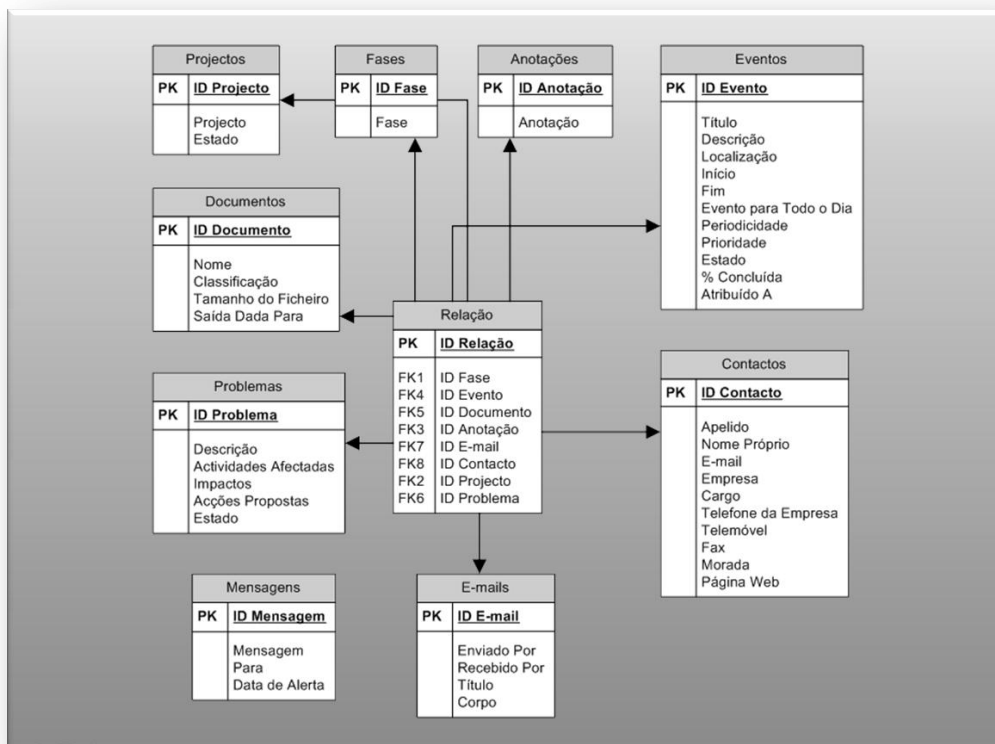


Figura 37 – Modelo de listas e suas relações

Os tipos de listas e colunas escolhidos para este modelo, apresentam-se nas tabelas seguintes.

Tabela 5 – Tipo de lista e colunas implementadas na lista de Projectos

Tipo de coluna	
Projectos (lista personalizada)	
ID Projecto	<i>inteiro</i>
Projecto	<i>texto</i>
Estado	<i>escolha</i>

Tabela 6 – Tipo de lista e colunas implementadas na lista de Fases

Tipo de coluna	
Fases (lista personalizada)	
ID Fase	<i>inteiro</i>
Fase	<i>texto</i>

Tabela 7 – Tipo de lista e colunas implementadas na lista de Eventos

Tipo de coluna	
Eventos (junção das listas de calendário e de tarefas)	
ID Evento	<i>inteiro</i>
Título	<i>texto</i>
Descrição	<i>texto avançado</i>
Localização	<i>texto</i>
Início	<i>data/hora</i>
Fim	<i>data/hora</i>
Evento para Todo o Dia	<i>selecção</i>
Periodicidade	<i>periodicidade</i>
Prioridade	<i>escolha</i>
Estado	<i>escolha</i>
% Concluída	<i>inteiro</i>
Atribuído A	<i>pessoa/grupo</i>

Tabela 8 – Tipo de lista e colunas implementadas na biblioteca de documentos Documentos

Tipo de coluna	
Documentos (biblioteca de documentos)	
ID Documento	<i>inteiro</i>
Nome	<i>texto</i>
Classificação	<i>escolha</i>
Tamanho do Ficheiro	<i>inteiro</i>
Saída Dada para	<i>pessoa/grupo</i>

Tabela 9 – Tipo de lista e colunas implementadas na lista de Anotações

Tipo de coluna	
Anotações (lista personalizada)	
ID Anotação	<i>inteiro</i>
Anotação	<i>texto avançado</i>

Tabela 10 – Tipo de lista e colunas implementadas na lista de E-mails

Tipo de coluna	
E-mails (lista personalizada)	
ID E-mail	<i>inteiro</i>
Enviado por	<i>texto</i>
Recebido por	<i>texto</i>
Título	<i>texto</i>
Corpo	<i>várias linhas de texto</i>

Tabela 11 – Tipo de lista e colunas implementadas na lista de Contactos

Contactos (lista de contactos)		Tipo de coluna
ID Contacto		<i>inteiro</i>
Apelido		<i>texto</i>
Nome Próprio		<i>texto</i>
E-mail		<i>texto</i>
Empresa		<i>texto</i>
Cargo		<i>texto</i>
Telefone da Empresa		<i>texto</i>
Telemóvel		<i>texto</i>
Fax		<i>texto</i>
Morada		<i>várias linhas de texto</i>
Página Web		<i>texto</i>

Tabela 12 – Tipo de lista e colunas implementadas na lista de Problemas

Problemas (lista personalizada)		Tipo de coluna
ID Problema		<i>inteiro</i>
Descrição		<i>várias linhas de texto</i>
Actividades Afectadas		<i>várias linhas de texto</i>
Impactos		<i>várias linhas de texto</i>
Ações Propostas		<i>várias linhas de texto</i>
Estado		<i>escolha</i>

Tabela 13 – Tipo de lista e colunas implementadas na lista de Relação

Relação (lista personalizada)		Tipo de coluna
ID Relação		<i>inteiro</i>
ID Projecto		<i>inteiro</i>
ID Fase		<i>inteiro</i>
ID Evento		<i>inteiro</i>
ID Documento		<i>inteiro</i>
ID Anotação		<i>inteiro</i>
ID E-mail		<i>inteiro</i>
ID Contacto		<i>inteiro</i>

Tabela 14 – Tipo de lista e colunas implementadas na lista de Mensagens

Mensagens (lista personalizada)		Tipo de coluna
ID Mensagem		<i>inteiro</i>
Mensagem		<i>texto avançado</i>
Para		<i>pessoa/grupo</i>
Data de Alerta		<i>data/hora</i>

6.3. Implementação

A implementação deste modelo encontra-se, até à presente data, numa fase embrionária pois apenas foram desenvolvidas Peças Web genéricas de acesso a listas e bibliotecas, de pesquisa e de implementação de ligações entre Peças Web. Estas duas últimas foram desenvolvidas numa primeira fase permitindo o refrescamento total da página e numa segunda fase com recurso à tecnologia AJAX. Todas estas Peças Web, expostas no Anexos A, ocultam o árduo trabalho necessário para se chegar até este ponto. Foram também abordados os *Event Handlers* que permitem detectar acções sobre listas e bibliotecas e alterar o seu comportamento.

Deste modo, é possível implementar a actualização em cascata nas listas que se pretendem, dando assim um comportamento semelhante ao de uma base de dados.

A partir destes elementos genéricos desenvolvidos, estão adquiridos os conhecimentos básicos de programação com acesso aos objectos do SharePoint, sendo agora possível implementar o modelo acima apresentado, garantindo todos os requisitos analisados no ponto 4, sem nunca descuidar o aspecto visual da interface e a sua usabilidade.

O investimento na exploração destas soluções personalizadas foi feito sempre com o intuito de melhorar a aplicação em termos de usabilidade. Por exemplo, o facto de se usar Peças Web ligadas com a utilização de AJAX permite que o utilizador trabalhe de forma rápida e simples pois todos os elementos presentes na interface apresentada na figura 34 são actualizados em função daquilo que o utilizador pretende, sem o refrescamento completo da página.

6.4. Síntese

A utilização apenas de objectos do SharePoint, com alguma personalização através do SharePoint Designer é uma solução fácil e rápida para o desenvolvimento de aplicações não muito complexas e com requisitos não muito exigentes em termos de apresentação visual. Quando se pretende ser mais exigente, nem o SharePoint nem nenhuma ferramenta permitem esse desenvolvimento rápido e fácil.

A utilização do SharePoint com a inclusão de soluções personalizadas desenvolvidas em Visual Studio oferecem um potencial de desenvolvimento enorme e bem mais rápido em relação ao desenvolvimento do portal inteiro a partir do nada. De facto, o que se desenvolve em código são apenas pequenas Peças Web que se adicionam à aplicação, não sendo necessário o desenvolvimento de mecanismos de autenticação, gestão de utilizadores e de permissões, nem mecanismos de controlo de versões de documentos, entre muitos outros exemplos de funcionalidades que o SP oferece já oferece de raiz.

Por exemplo, o código implementado na Peça Web que acede a uma lista, cujo exemplo se encontra no Anexo A, apenas transfere todos os elementos presentes na lista para um objecto de visualização, sem qualquer preocupação a nível das permissões do utilizador autenticado. No entanto, quando se utiliza essa Peça Web na página, os resultados apresentados, para a mesma lista, são diferentes dependendo das permissões do utilizador autenticado, facto que é gerido internamente pelo SharePoint.

O modelo apresentado dá, aparentemente, garantias de um funcionamento correcto numa interface que proporcionasse uma boa experiência de utilização, tal como era pretendido. Este modelo responde assim a todos os requisitos básicos para o funcionamento experimental do sistema de gestão de projectos. Até ao momento, o teste deste modelo foi feito apenas com base num exemplo muito simples, faltando por isso a construção de um protótipo para um teste mais exaustivo. O recurso a soluções desenvolvidas em código foi determinante para este modelo, sendo até imprescindível. No entanto estas soluções trazem como principais desvantagens a manutenção futura da aplicação e o desaproveitamento de eventuais funcionalidades não exploradas do SP.

Capítulo 7

Conclusões

Neste capítulo encontram-se as principais conclusões relativas ao trabalho desenvolvido e são apresentadas algumas perspectivas de desenvolvimento futuro.

7.1. Principais Conclusões

O estudo efectuado sobre as ferramentas de gestão de projectos existentes no mercado revelou que, para esta aplicação, essas ferramentas apresentam uma insuficiência de funcionalidades, principalmente a nível da agilidade e flexibilidade.

Hoje em dia, os sistemas apresentados para a gestão de projectos não respondem às necessidades reais de um gestor por serem demasiado rígidos e estruturados, não dando espaço a projectos não estruturados nem liberdade de escolha ao utilizador.

Estes sistemas encontram-se, normalmente, muito vocacionados para o planeamento do projecto e sua gestão em termos de tempo, custos e recursos. A componente colaborativa é habitualmente esquecida e a integração com o e-mail ainda é uma miragem, na maior parte dos casos.

Há contudo alguma preocupação com o aspecto gráfico dos sistemas mas, no que respeita à usabilidade e dinamismo oferecidos pela interface da aplicação, este tipo de sistemas ficam muito aquém. As informações presentes na interface apresentam-se como módulos individuais, sendo necessário entrar em cada um para se ter acesso a essas informações.

O mesmo se passa em relação a cada projecto porque não há uma vista agregada com o resumo dos últimos desenvolvimentos, sendo também necessário entrar em cada um para ver as últimas actualizações.

Em suma, estes sistemas não se preocupam com os aspectos humanos e, na sua maioria, são demasiado complexos para o utilizador comum. Por estes motivos, muitos gestores de projectos continuam a fazer da sua aplicação de e-mail ou de um blogue pessoal a ferramenta

de eleição para gerirem os seus projectos. Tal situação é, todavia, incomportável para projectos de maior dimensão.

Os novos paradigmas sobre os métodos de trabalho e as formas de interacção entre as pessoas, descritos no capítulo 3, ajudaram a compreender melhor como as pessoas realmente trabalham e interactuam entre si e de que forma os sistemas de informação podem auxiliá-las.

Neste sentido, o HIM considera cinco funcionalidades importantes e necessárias, suportadas por seis objectos, para que a gestão das interacções humanas resulte de forma eficiente e produza resultados para as organizações.

Resumindo, os sistemas actuais devem permitir a visibilidade entre utilizadores, para que cada um saiba quem faz o quê, e deve fornecer uma estrutura de mensagens entre os mesmos.

Devem também oferecer um suporte para o trabalho mental que as pessoas realizam e que muitas vezes é ignorado por ser difícil de medir.

Por fim, a gestão das actividades deve ser flexível em vez de descritiva e autoritária, uma vez que as pessoas não estão com o mesmo humor todos os dias, logo não têm a mesma capacidade todos os dias, e por conseguinte, os processos devem também ser flexíveis para permitirem a alteração de outros processos.

O GTD sugere um modelo que ajuda as pessoas a organizarem o seu trabalho e a decidirem o que têm de fazer a seguir. Neste sentido, é sugerido que as pessoas libertem a sua mente de informações desnecessárias e as guardem em sistemas de memória externos sob a forma de acções, caso isso seja possível.

Isto tem particular interesse na forma como se desenvolve uma aplicação pois esta deve orientar o utilizar a introduzir informações sob a forma de acções.

O State-Flow aceita a forma de trabalhar das pessoas, não as obrigando à sequência rígida de execução de actividades, como é típico num fluxo de trabalho. Desta forma, é criada a noção de estado do processo onde o conjunto de actividades que se desenvolvem nesses estados cria um fluxo que converge para a conclusão do processo, ou seja, o objectivo pretendido.

Na realidade, as pessoas trabalham dessa forma pois vão fazendo as actividades sem nenhuma sequência rígida e o conjunto dessas actividades realizadas pelas pessoas faz com que o trabalho esteja mais perto do final.

O Activity Centric sugere aplicações com interfaces para o utilizador bem desenhadas, com um aspecto agradável e que lhes ofereça os serviços que precisam num dado contexto, isto é, que seja intuitiva e fácil de usar.

Nestas interfaces têm que estar disponíveis para o utilizador todos os elementos que este precisa num dado contexto. Contudo, deve-se ter atenção à quantidade destes elementos e à forma como estes aparecem dispostos na interface, pois só dificulta o trabalho das pessoas se forem apresentadas demasiadas informações ou se estas não tiverem uma lógica de organização.

Estas ideias ajudaram a perceber melhor como as pessoas trabalham e interactivam entre si e que funcionalidades o sistema deveria incluir. O passo seguinte foi estudar o SP e analisar qual a melhor forma de implementar essas funcionalidades.

Após a análise de requisitos, houve alguma dificuldade para se chegar a um consenso em relação às entidades a criar porque estão todas muito ligadas. Por exemplo, a entidade Actividade, que permite a criação de tarefas, não deixa de ser uma comunicação e um registo.

Dada intenção de explorar e analisar o SharePoint como ferramenta de suporte ao desenvolvimento de várias aplicações e, em particular, desta aplicação, houve um esforço nesse sentido que atravessou diferentes etapas, o que acabou por consumir bastante tempo. Todavia, esse tempo dispendido e esse esforço empregue não foi em vão dado que, no final do projecto, se criou uma ideia mais sólida e fundamentada na experiência prática sobre o que o SharePoint oferece, o que é possível fazer, com que meios e em que tempo.

Desta forma, agora é possível tirar conclusões sobre a melhor forma de desenvolver uma aplicação em SP, em função dos requisitos desta.

A caminhada até este ponto foi difícil e recheada de dificuldades, tanto mais que os conhecimentos eram praticamente nulos, como já foi referido, mas, no entanto, isso sempre foi encarado como um estímulo para continuar a explorar o SP.

Muitos pormenores do SP, por vezes importantes ou até imprescindíveis, não se encontram presentes nos livros mas apenas em fóruns da comunidade Microsoft.

Ao longo do projecto foram testadas várias estratégias para a implementação em SP o que nos permitiu um conhecimento bastante profundo sobre a ferramenta.

Estas várias fases de testes do SharePoint passaram pelo desenvolvimento de *User Controls* e a utilização de uma base de dados externa, depois pela utilização apenas de objectos do SP e personalização através do SharePoint Designer e por fim, pelo desenvolvimento de Peças Web em Visual Studio que acedem aos objectos do SharePoint.

Destas três distintas abordagens, a mais eficaz é claramente a última, uma vez que dá liberdade total para se personalizar tudo.

Nesta última fase surgiu algumas vezes a dúvida: "Porquê utilizar então o SharePoint e não desenvolver tudo em código?". A resposta é simples: há muitas funcionalidades que o SP oferece e que seriam árduas e morosas de implementar, como por exemplo o sistema de gestão de permissões, o controlo de versões, os tipos de conteúdo, etc.

Porém, para aplicações com outro tipo de requisitos, o desenvolvimento através do browser com alguma personalização através do SharePoint Designer pode fazer sentido, não se justificando o desenvolvimento de código específico. Por exemplo, uma empresa que pretenda publicar informações na Web e precise de um site que permita facilmente criar páginas com editores de conteúdos e permita submetê-las para aprovação antes de serem publicadas, neste caso não se justifica qualquer desenvolvimento extra pois o SP já oferece estas funcionalidades todas. Em suma, tudo depende do tipo de aplicação.

Na perspectiva de decisão entre a utilização do WSS ou do MOSS, uma vez que o MOSS implica a obtenção de uma licença, enquanto que o WSS é livre, aqui, mais uma vez a resposta é simples, pois tudo depende das funcionalidades que se pretendem implementar, com que esforço e com que tempo. Obviamente, o WSS oferece menos funcionalidades mas para uma dada aplicação podem ser suficientes.

O modelo apresentado no capítulo 6 dá, aparentemente, garantias de um funcionamento correcto numa interface que proporcionasse uma boa experiência de utilização, tal como era pretendido. Este modelo responde assim a todos os requisitos básicos para o funcionamento experimental do sistema de gestão de projectos.

A implementação da totalidade dos requisitos, especificados no capítulo 4, foi pensada para ser gradual num futuro próximo. Estas evoluções que a ferramenta iria sofrer seriam também baseadas nas experiências vividas pelos utilizadores e nos eventuais novos requisitos que poderiam surgir.

O recurso a soluções desenvolvidas em código foi determinante para este modelo, sendo até imprescindível. No entanto estas soluções trazem como principais desvantagens a manutenção futura da aplicação e o desaproveitamento de eventuais funcionalidades do SP não exploradas.

A implementação deste modelo encontra-se, até à presente data, numa fase embrionária porque, dado o tempo dispendido na exploração do SP, apenas foram desenvolvidas Peças Web genéricas.

Este tipo modelo é um ponto de partida para outras aplicações que precisem de um modelo idêntico a uma base de dados. Contudo, este pode ainda ser melhorado se forem utilizados os identificadores únicos do próprio SP e não apenas da lista, o que reduziria a lista de relações para apenas duas colunas, para além da coluna com ID dos itens da lista.

7.2. Perspectivas de desenvolvimento

Neste sistema, poderia evoluir com a inclusão de novas funcionalidades como, por exemplo, receber os e-mails directamente na aplicação de gestão de projectos em vez de receber na aplicação cliente de e-mail, incluir indicadores estatísticos mais completos, contemplando já o controlo da qualidade e proporcionar também a gestão de recursos.

Se fosse sentida essa necessidade, poderia incluir-se a funcionalidade de RSS, embora, este deva ser um sistema de consulta diária pelo que, aparentemente, não se justifique tal inclusão.

Poderiam também ser acrescentadas mais áreas colaborativas como Wikis, áreas de debate ou blogues, com o risco de serem abandonadas as outras funcionalidades já existentes por parte dos gestores.

No futuro, seria também interessante acrescentar ao sistema um mecanismo de gestão da comunicação por voz e por videoconferência, dada a importância e o crescimento que estas tecnologias têm sofrido nos últimos anos.

Em relação ao SharePoint, para este sistema, era conveniente explorar os Excel Services e os KPI para se tirar mais partido do Excel e fornecer outro tipo de indicadores estatísticos. Seria também interessante explorar o Microsoft Office Project Server 2007 para se integrar de forma eficaz as funcionalidades do Microsoft Project no SharePoint.

Quanto às funcionalidades em geral, há muito ainda por descobrir no SharePoint, razão pela qual não se sugere aqui nenhuma.

Referências

- [1] Project Management Institute, A Guide to the Project Management Body of Knowledge, Project Management Institute, 2000.
- [2] Mark Brown, A Gestão de Projectos com Sucesso, Presença, 1992.
- [3] Microsoft Office Project 2007, disponível em [http://technet.microsoft.com/pt-br/library/cc197654\(en-us\).aspx](http://technet.microsoft.com/pt-br/library/cc197654(en-us).aspx), acessado a última vez em 30 de Junho de 2008.
- [4] @Task, disponível em <http://www.attask.com/>, acessado a última vez em 30 Junho 2008.
- [5] IBM, Activity Explorer: Activity-centric collaboration from research to product, disponível em <http://www.research.ibm.com/journal/sj/454/geyer.html>, 2006, acessado a última vez em 30 Junho 2008.
- [6] Helsinki University of Technology, Human Interaction Management – Adding Human Factors into Business Processes Management, disponível em http://www.soberit.hut.fi/T-86/T-86.5161/2006/HIM_Humanedj_final.pdf, acessado a última vez em 30 de Junho de 2008.
- [7] Peter Fingar, Extreme Competition: The Greatest Innovation since BPM, disponível em http://www.bptrends.com/deliver_file.cfm?fileType=publication&fileName=SIX%2D03%2D07%2DCOL%2DTheGreatestInnovationSinceBPM%2DFingar%2DFinal1%2Epdf , 2007, acessado a última vez em 30 Junho 2008.
- [8] Keith Harrison-Broninski, Human Interactions: The Heart And Soul Of Business Process Management: How People Really Work And How They Can Be Helped To Work Better, Meghan-Kiffer Press, 2005.
- [9] David Allen, Getting Things Done: The Art of Stress-Free Productivity, Penguin Books, 2001.
- [10] Francis Heylighen, Getting Things Done: The Science Behind Stress-Free Productivity, disponível em <http://pespmc1.vub.ac.be/Papers/GTD-cognition.pdf>, acessado a última vez em 30 de Junho de 2008.
- [11] Ilia Bider, State-Flow Technique for Business Process Analysis: Case Studies, 2002, disponível em <http://www.emeraldinsight.com/Insight/viewPDF.jsp?Filename=html/Output/Published/EmeraldFullTextArticle/Pdf/0880150103.pdf>, acessado a última vez em 30 Junho 2008.

- [12] IBM, Business activity patterns: A new model for collaborative business applications, disponível em <http://www.research.ibm.com/journal/sj/454/moody.html>, 2006, acedido a última vez em 30 Junho 2008.
- [13] Ilia Bider, Aligning People to Business Processes Experience Report, 2005, disponível em <http://www3.interscience.wiley.com/cgi-bin/fulltext/112177291/PDFSTART>, acedido a última vez em 30 de Junho de 2008.
- [14] David I. Cleland, Roland Gareis, Global Project Management Handbook, 2nd ed., McGraw Hill, 2006.
- [15] Rory Burke, Project Management 3rd ed., John Wiley & Sons, 2001.
- [16] Richard Murch, Project Management: Best Practices for IT Professionals, Prentice-Hall, 2001.
- [17] Human Interaction Management System, disponível em <http://human-interaction-management.info/>, acedido a última vez 30 Junho 2008.
- [18] IBM, Beyond predictable workflows: Enhancing productivity in artful business processes, disponível em <http://www.research.ibm.com/journal/sj/454/hill.html>, 2006, acedido a última vez em 30 Junho 2008.
- [19] David Allen, Ready for Anything: 52 Productivity Principles for Work and Life, Viking, 2003.
- [20] Keith Harrison-Broninski, Modeling Human Interactions: Part 1, 2005, disponível em <http://www.businessprocesstrends.com/publicationfiles/06-05%20WP%20Modeling%20Human%20Interactions%20-%20Harrison-Broninski.pdf>, acedido a última vez em 30 Junho 2008.
- [21] Keith Harrison-Broninski, Modeling Human Interactions: Part 2, 2005, disponível em <http://www.businessprocesstrends.com/publicationfiles/07-05%20WP%20Modeling%20Human%20Interactions%20-%20Pt%202%20-%20Harrison-Bro%E2%80%A6.pdf>, acedido a última vez em 30 Junho 2008.
- [22] Scot Hillier, Beginning ASP.NET 3.5 in C# 2008 From Novice to Professional, 2nd ed., Apress, 2007
- [23] Scot Hillier, Microsoft SharePoint Building Office 2007 Solutions in C# 2005, Apress, 2007
- [24] Wrox, Professional Sharepoint 2007 Development, 2007
- [25] Wrox, SharePoint 2007 and Office Development Expert Solutions, 2007
- [26] Microsoft Press, 7 Development Projects for Office SharePoint Server 2007 and Windows SharePoint Services Version 3.0, 2006.
- [27] Microsoft Press, Microsoft Office SharePoint 2007 – Administrator's Companion, 2007.
- [28] Microsoft Sharepoint, disponível em <http://office.microsoft.com/pt-pt/sharepointserver/FX100492002070.aspx> , acedido a última vez em 30 Junho 2008.
- [29] Jan Tielens, SharePoint MVP, blog <http://weblogs.asp.net/jan/default.aspx> , acedido a última vez em 30 Junho 2008.
- [30] Liam Cleary, SharePoint MVP, blog <http://www.helloitsliam.com/default.aspx> , acedido a última vez em 30 Junho 2008.

[31] Zlatan, SharePoint MVP, blog <http://dotnet.org.za/zlatan/default.aspx>, acedido a última vez em 30 Junho 2008.

[32] Daniel Larson, blog

<http://forums.microsoft.com/forums/ShowPost.aspx?PostID=3452511&SiteID=1>, acedido a última vez em 30 de Junho 2008.

[33] João Silva, Tese: “Sistema de Gestão de Equipamentos e da Manutenção”, FEUP, 2008.

Anexos

A. Soluções Personalizadas para SharePoint

Centro de Administração

Após a instalação do MOSS 2007 é necessário efectuar algumas configurações iniciais no Centro de Administração, que não é mais que um site de SharePoint. As tarefas mais comuns são automaticamente apresentadas ao administrador no portal de administração, sob forma de tarefas, tal como ilustrado na figura 38.

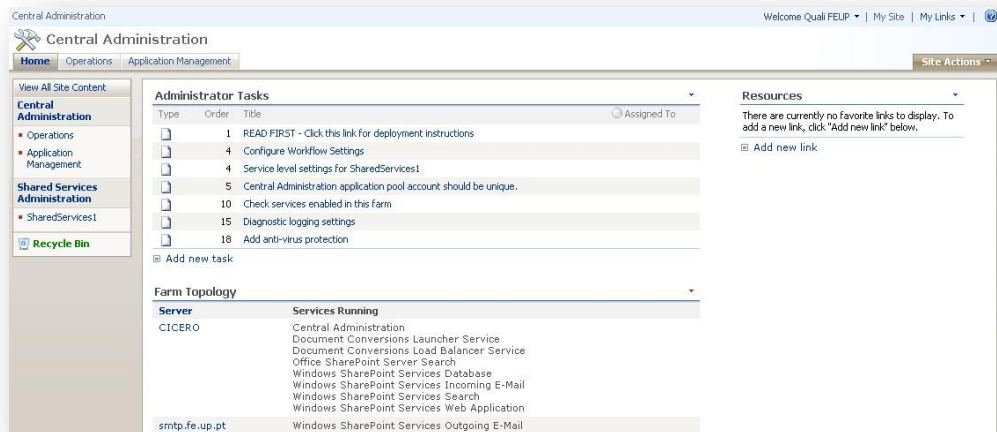


Figura 38 – Página inicial do Centro de Administração

As acções mais comuns aqui realizadas passam pela gestão de Peças Web, gestão de *Web Applications*, *IIS Pools*, que estruturam as referidas *Web Applications*, e gestão de *Site Collections*, conjunto hierárquico de sites. A integração de *Web Applications* e *Site Collections* está ilustrada na figura 39.

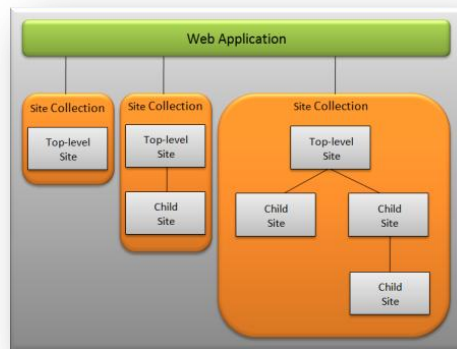


Figura 39 – Arquitetura da *Web Application*

Em *Operations* pode-se encontrar tarefas de configuração agrupadas em oito categorias, tal como ilustrado na figura 40.

Central Administration > Operations

Operations

This page contains links to pages that help you manage your server or server farm, such as changing the server farm topology, specifying which services are running on each server, and changing settings that affect multiple servers or applications.

<p>Topology and Services</p> <ul style="list-style-type: none"> ▣ Servers in farm ▣ Services on server ▣ Outgoing e-mail settings ▣ Incoming e-mail settings ▣ Approve/reject distribution groups 	<p>Global Configuration</p> <ul style="list-style-type: none"> ▣ Timer job status ▣ Timer job definitions ▣ Master site directory settings ▣ Site directory links scan ▣ Alternate access mappings ▣ Manage farm features ▣ Quiesce farm ▣ Solution management
<p>Security Configuration</p> <ul style="list-style-type: none"> ▣ Service accounts ▣ Information Rights Management ▣ Antivirus ▣ Blocked file types ▣ Update farm administrator's group ▣ Information management policy configuration ▣ Manage settings for single sign-on 	<p>Backup and Restore</p> <ul style="list-style-type: none"> ▣ Perform a backup ▣ Backup and restore history ▣ Restore from backup ▣ Backup and restore job status
<p>Logging and Reporting</p> <ul style="list-style-type: none"> ▣ Diagnostic logging ▣ Usage analysis processing ▣ Information management policy usage reports 	<p>Data Configuration</p> <ul style="list-style-type: none"> ▣ Default database server ▣ Data retrieval service
<p>Upgrade and Migration</p> <ul style="list-style-type: none"> ▣ Microsoft Content Management Server migration ▣ Enable Enterprise Features ▣ Enable features on existing sites ▣ Convert license type 	<p>Content Deployment</p> <ul style="list-style-type: none"> ▣ Content deployment paths and jobs ▣ Content deployment settings ▣ Check deployment of specific content

Figura 40 – Página de *Operations* do Centro de Administração

Aqui, destacam-se os seguintes grupos:

Topology And Services

Configuração dos servidores e das definições de e-mail

Global Configuration

Para gestão dos pacotes de soluções que podem ser instalados no servidor e configuração do mapa de endereços dos sites presentes no servidor

Security Configuration

Para a gestão da segurança do SP, como por exemplo a gestão dos tipos de ficheiros bloqueados pelo SP

Backup and Restore

Utilizado para efectuar copias de segurança e restauro de sites, bem como, consultar o histórico de acções e o estado actual da operação de backup ou de restauro.

A imagem 41 ilustra as funcionalidades presentes em *Application Management*.

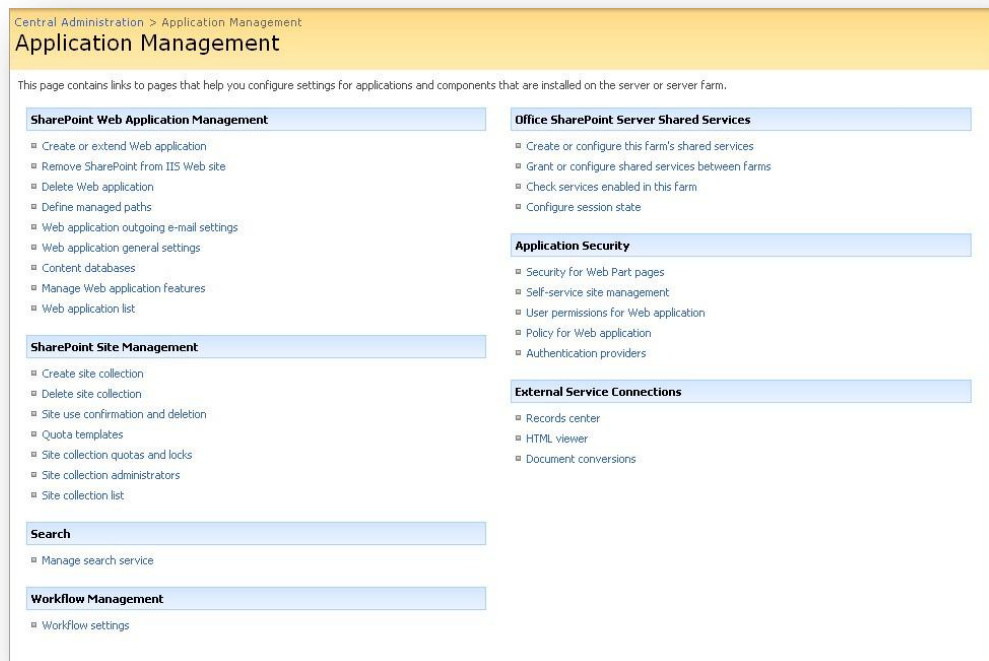


Figura 41 – Página de *Application Management* do Centro de Administração

Aqui, destacam-se os seguintes grupos:

Sharepoint web Application Management

Para gestão de Web Applications

Sharepoint Site Management

Para gestão de Site Collections

Application Security

Para a gestão das permissões de acesso, ao nível das Web Applications, como por exemplo, activar ou desactivar o acesso anónimo

Acesso Anónimo

O acesso anónimo, quando activo para um dado site, permitido aos utilizadores anónimos, assim como aos utilizadores autenticados aos quais não foi concedido acesso ao site, acederem a todo o site, incluindo qualquer lista, biblioteca, pasta, item ou documento de uma lista ou biblioteca que herde as respectivas permissões desse site.

O acesso anónimo pode ser configurado a três níveis:

- Concedendo acesso anónimo a um site inteiro.
- Concedendo acesso anónimo apenas a listas e bibliotecas.
- Bloqueando acesso anónimo a um site.

Pesquisa

Por predefinição, o serviço de pesquisa encontra-se desactivo no WSS 3.0, sendo necessário configurá-lo em *SharedServices1*. Aqui, define-se as fontes de conteúdos que vão ser indexadas, a sua periodicidade de actualização e o tipo de indexação, incremental ou completa, tal como se pode ver na figura 42.

Shared Services Administration: SharedServices1 > Search Settings > Content Sources > Edit Content Source

Edit Content Source

Use this page to edit a content source.
* Indicates a required field

Name Type a name to describe this content source.	Name: * Local Office SharePoint Server sites
Content Source Details This shows the current status of the Content Source.	Content Source Type: SharePoint Sites Current Status: Idle Current Number of Start Addresses: 2 Last Crawled At: 5/26/2008 10:23 AM Errors in Last Crawl: 0
Start Addresses Type the URLs from which the search system should start crawling.	Type start addresses below (one per line): * http://cicero:5/sps3//cicero:5/ Example: http://intranetsite
Crawl Settings Specify the behavior for crawling this type of content. Selecting to crawl everything under the hostname will also crawl all the SharePoint Sites in the server.	Select crawling behavior for all start addresses in this content source: <input checked="" type="radio"/> Crawl everything under the hostname for each start address <input type="radio"/> Crawl only the SharePoint Site of each start address
Crawl Schedules Select the crawl schedules for this content source.	Select the schedule that this should be a part of: Full Crawl Every 20 minute(s) from 16:52 for 24 hour(s) every day, starting 23-04-2008 Edit schedule Incremental Crawl None Create schedule
Start Full Crawl Select "Start full crawl of this content source" and click "OK" to start a full crawl of this content source.	<input checked="" type="checkbox"/> Start full crawl of this content source

Figura 42 – Página de configuração da pesquisa no Centro de Administração

Aqui é também possível iniciar o serviço de indexação manualmente, assim como consultar os registos de actividade, entre outras operações.

O Comando STSADM

O MOSS 2007 inclui uma ferramenta de administração para ser utilizada na linha de comandos, que é o comando stsadm. Este, para além de todas as operações possíveis de realizar no Centro de Administração, permite ainda efectuar outras não contempladas neste, como por exemplo, mudar a porta utilizada pelo portal de administração.

Este comando encontra-se localizado na pasta de instalação do SP e é bastante útil para criar tarefas periódicas, como por exemplo as cópias de segurança, embora sejam necessárias permissões de administrador para o poder utilizar.

Cópia de Segurança e Restauo

O MOSS 2007 permite fazer cópias de segurança a vários níveis e de diferentes formas. É então possível fazer um backup da totalidade dos sites de SharePoint, assim como também é possível fazer o backup de um site específico ou apenas do seu conteúdo.

A cópia de segurança pode ser diferencial, que inclui apenas os novos conteúdos, em relação ao último backup, ou completa, que inclui todos os conteúdos.

Esta, pode ser feita através do Centro de Administração do SharePoint, no grupo *Backup and Restore* que se encontra em *Operations*, como também através do comando stsadm, onde se também se encontra todas estas opções disponíveis. Por exemplo, para se efectuar um backup de um site, pode-se introduzir o comando:

```
stsadm -o backup -url <url> -filename <nome do ficheiro>
```

Onde, <url> corresponde ao endereço do site e <nome do ficheiro> corresponde ao caminho e nome do ficheiro que conterà os dados da cópia de segurança.

Do mesmo modo, para se efectuar um restauro a partir de um backup, pode-se introduzir o comando:

```
stsadm -o restore -url <url> -filename <nome do ficheiro>
```

Gestão de Peças Web

Quando se pretende instalar uma solução, para que esta esteja disponível no SP, como por exemplo, uma Peça Web personalizada, é necessário primeiro efectuar a sua instalação, com recurso ao comando:

```
stsadm -o addsolution -filename <nome do ficheiro>
```

Seguidamente, deve-se proceder à sua disponibilização nas *Web Applications* pretendidas, recorrendo para tal ao *Solution Management* disponível no Centro de Administração. Por fim, é preciso activar a sua funcionalidade nos portais pretendidos, na configuração do respectivo portal.

Personalização

O MOSS 2007 apresenta de facto inúmeras funcionalidades por predefinição tal como já foi enunciado no início deste capítulo. Contudo, há ainda dois níveis de personalização que permitem construir soluções à medida para cada aplicação.

O primeiro nível de personalização encontra-se dividido entre configurações que podem ser realizadas directamente através do browser e outras que são feitas com recurso ao software SharePoint Designer 2007.

O segundo nível passa pelo desenvolvimento de código em Visual Studio, para soluções mais complexas e de nível mais profissional, permitindo elevar significativamente as potencialidades do SP.

A figura 43 mostra o enquadramento dos níveis de personalização, com a estrutura física de servidores, num cenário de desenvolvimento e produção.

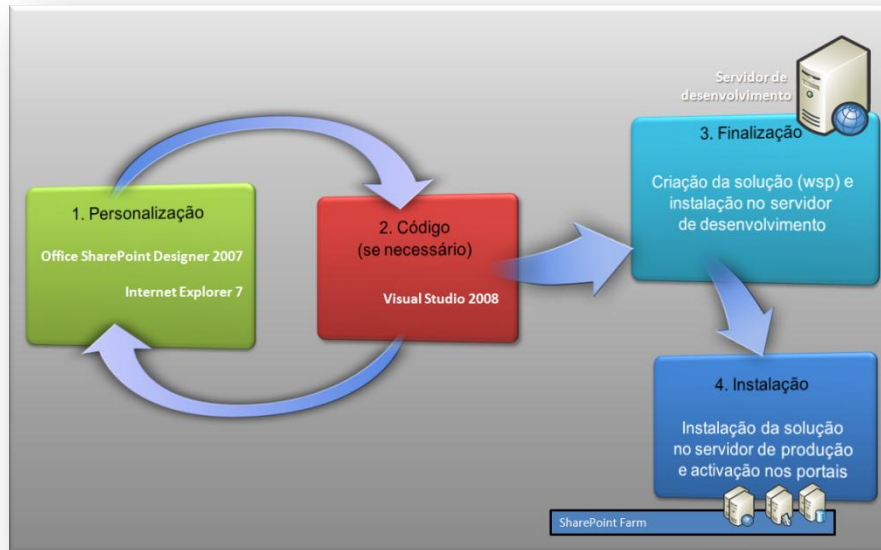


Figura 43 – Cenário de desenvolvimento e produção

Estrutura de um Portal em SharePoint

Em primeiro lugar, importa perceber a estrutura de um portal em SP, que basicamente é constituído por uma hierarquia de três elementos principais, o site de topo, sub-sites e páginas.

Na figura 44 pode-se ver a estrutura de um site e a forma como este é visualizado pelos utilizadores. São claramente identificáveis os três elementos já referidos, pelo que, é fácil compreender a organização hierárquica de um site em SharePoint.

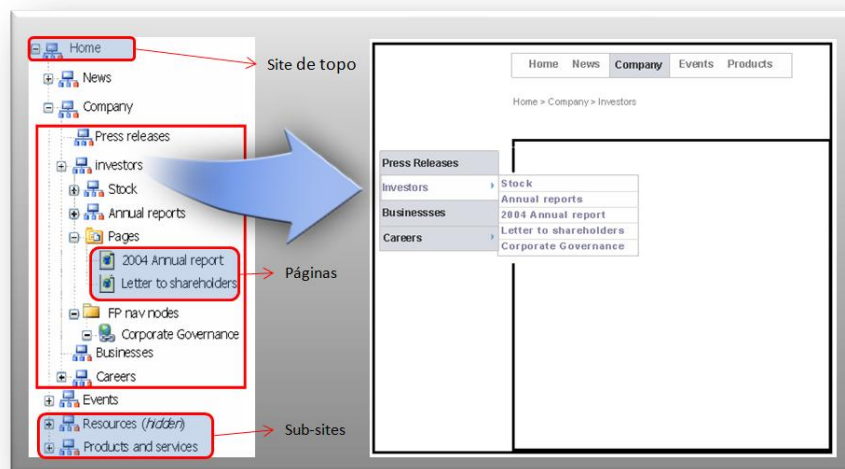


Figura 44 – Mapeamento dos objectos do SharePoint para o site

Página Mestre e Esquema de Página

Cada página é por sua vez constituída pela conjugação de uma página mestra, *Master Page*, que é comum a todo o site e, tipicamente, define a forma como a navegação é mostrada, e um esquema de página, *Page Layout*, que define a forma como o conteúdo é organizado e mostrado. O *Page Layout* define basicamente as Zonas de Peças Web, *Web Part Zones*, onde estas podem ser adicionadas.

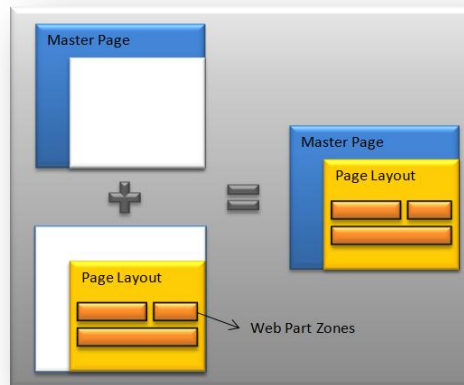


Figura 45 – Relação entre *Master Page* e *Page Layout*

A personalização efectiva da *Master Page*, *Page Layout* e *Web Part Zones* é feita no SharePoint Designer sendo que, pelo browser, é possível aplicar diferentes *Master Pages*, mudar o tema do site, que corresponde à alteração das folhas de estilo, CSS, e alterar a posição das Peças Web, movendo-as entre as *Web Part Zones*.

Microsoft Office SharePoint Designer 2007

A principal vocação do SharePoint Designer 2007 é permitir criar e implementar soluções para a plataforma SharePoint, sem que seja necessário escrever código. A imagem 46 mostra o ambiente de desenvolvimento desta ferramenta.

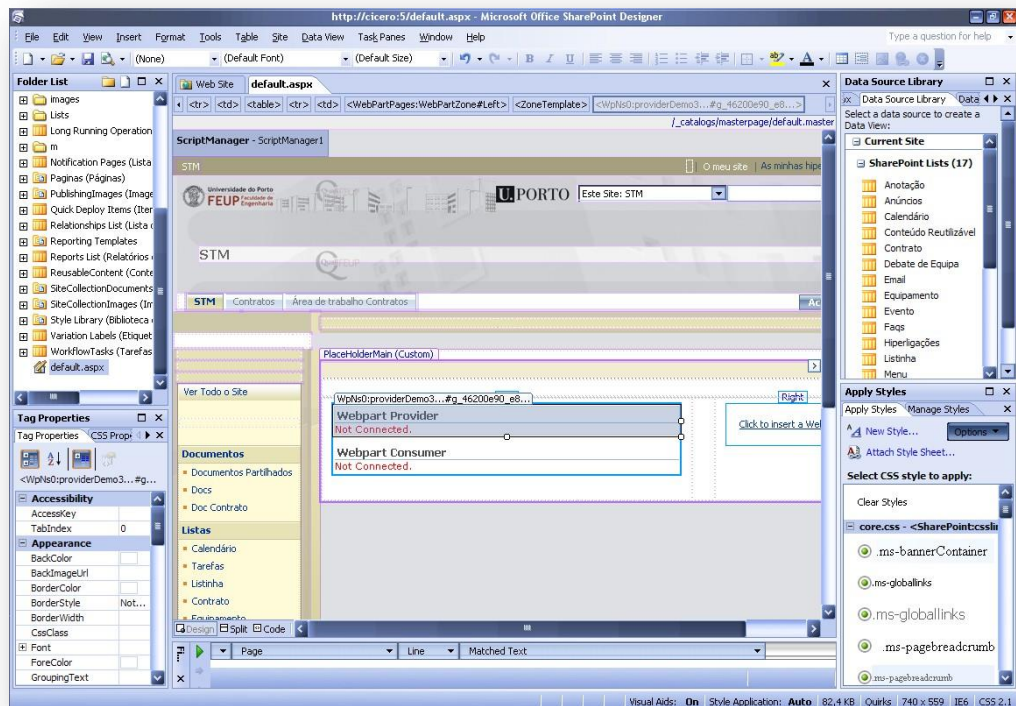


Figura 46 – Interface do SharePoint Designer

Estrutura de Pastas do SharePoint Designer

A figura 47 pretende ilustrar tanto o conteúdo como a estrutura principal de um portal SP, que aparece na lista de pastas do SharePoint Designer. Os elementos realçados permitem perceber o modo de armazenamento de conteúdos e a forma como o SP organiza o sistema de ficheiros associado.

Na biblioteca de estilos, denominada por "Style Library", encontram-se as folhas de estilo, CSS, e a pasta contendo as definições predefinidas de estilos visuais possíveis de aplicar às vistas de Peças Web.

Na pasta "Lists" são guardadas todas as listas criadas pelos utilizadores do portal e também aquelas criadas por defeito pelo SP, como por exemplo, a lista de tarefas, calendário, hiperligações, etc. Dentro de cada uma destas listas encontram-se ainda as páginas associadas a cada uma, como por exemplo, a página de visualização dos itens, AllItems.aspx, a página de inserção de novos itens, NewForm.aspx, etc.

Aqui, é possível definir outras páginas associadas à lista, através das propriedades da mesma, tal como se pode ver na figura 48.

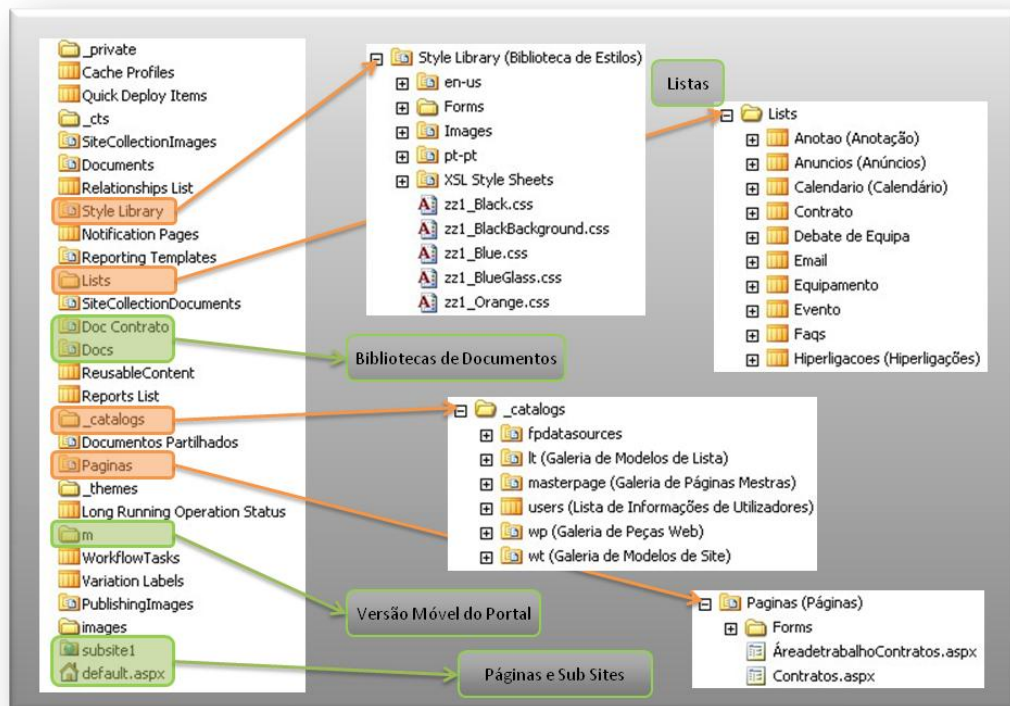


Figura 47 – Relação da arquitectura de pastas do SharePoint Designer com os objectos do SharePoint

Ao contrário das listas, as bibliotecas são todas guardadas ao nível da pasta "Lists" e a sua principal função é servir como repositório de documentos e/ou páginas.

A pasta "_catalogs" reúne as informações sobre as páginas mestras disponíveis, utilizadores e Peças Web do portal, entre outras.

A informação sobre versão móvel do portal, referida no ponto 5.4.1, encontra-se armazenada na pasta "m".

Os sub-sites são igualmente guardados em pastas que contêm uma estrutura de organização idêntica à já referida nos parágrafos anteriores.

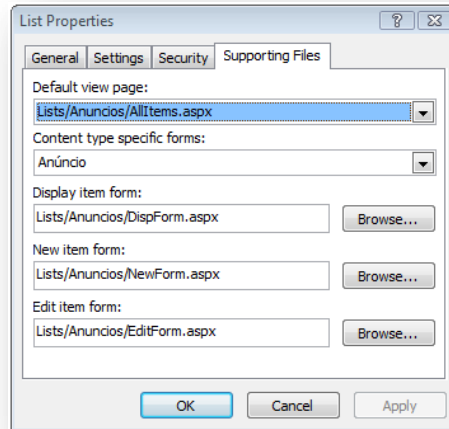


Figura 48 – Quadro de propriedades de uma lista

Personalização de uma Página Mestre

A título de exemplo, a imagem 49 permite perceber a personalização efectuada numa *Master Page*, onde a parte superior foi remodelada para cumprir os requisitos de um dado site.

Esta alteração exemplifica a facilidade com que um portal pode ser personalizado em termos visuais dando resposta a qualquer requisito neste âmbito.

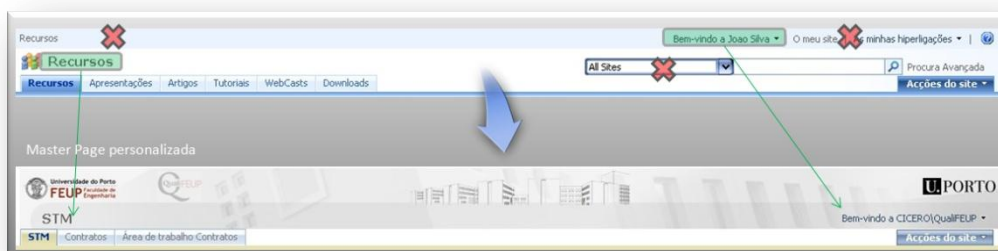


Figura 49 – Exemplo de personalização de uma *Master Page*

Quanto às alterações ao *Page Layout*, basicamente é possível definir novas zonas onde poderão ser inseridas Peças Web e, definindo as suas dimensões, é possível gerir o aspecto do site de forma coerente, facilitando a sua manutenção.

Formulários

O SP cria por predefinição os formulários para a introdução de itens para todas as listas ou bibliotecas existentes, cujo aspecto é de difícil manutenção em aplicações com requisitos um pouco mais específicos.

A figura 50 mostra o que facilmente se pode conseguir alterar sem que para isso seja necessário qualquer programação, bastando apenas a utilização do *Custom List Form*.

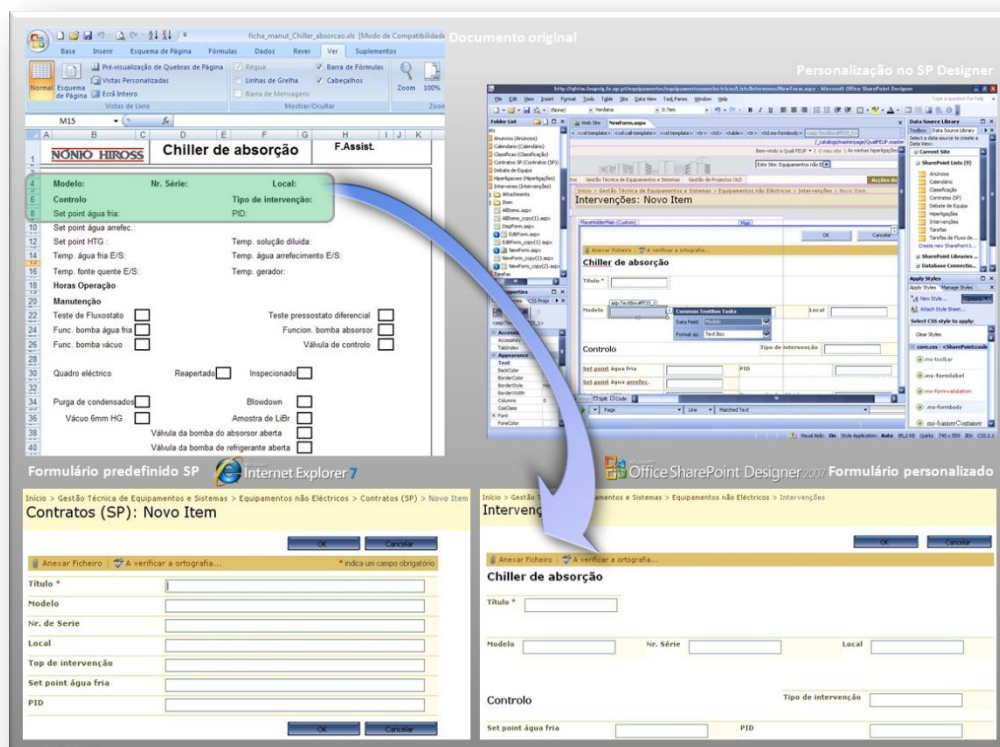


Figura 50 – Exemplo de formulários personalizados no SharePoint

Utilização de Custom List Form

Para se efectuar uma personalização nos formulários criados pelo SP é necessário editar a página *NewForm.aspx* da lista que se pretende alterar com o SP Designer. Em primeiro lugar, é necessário seleccionar o formulário, que não é mais que uma *Peça Web*, e apagá-lo. Seguidamente, na opção *Custom List Form*, que se encontra em *SharePoint Controls* do menu *Insert*, é possível definir qual a lista que serve de base para este formulário, indicar o tipo de conteúdo usado e especificar que tipo de formulário se pretende criar, se é um formulário de edição, visualização ou inserção de itens, tal como se pode ver na figura 51.



Figura 51 – Quadro de inserção de um novo formulário personalizado

Assim que este formulário personalizado estiver inserido, é possível modificar o aspecto visual na sua totalidade com alterações de tabelas, imagens, tamanho de letra, disposição dos campos, etc.

De notar que, para alterar os componentes de inserção de informação por parte dos utilizadores é necessário alterar o tipo de campo do respectivo componente. A figura 52 mostra a alteração do componente onde é inserido o título do item, que inicialmente está formatado como "*List From Field*" e passa para uma "*Text Box*" onde se pode definir as suas propriedades, como por exemplo a dimensão, alinhamento, classe, etc.



Figura 52 – Campos de inserção de um novo formulário personalizado

Ao substituir o formulário original por um personalizado este passa a ter um comportamento estático relativamente ao original porque, por exemplo, uma alteração de uma coluna na lista

apenas se reflecte de forma automática nos formulários predefinidos, sendo este um preço a pagar quando se opta por personalização em SP a este nível.

XSLT

Outro exemplo do que é possível fazer no Designer refere-se à utilização de XSLT, linguagem baseada na XML, usada na transformação de informação XML. O aspecto que as Peças Web apresentam pode também ser alterado completamente transformando cada uma destas numa XSLT Data View. Com esta operação, algumas funcionalidades nas propriedades da Peça Web deixam de estar disponíveis quando a página se encontra em modo de edição. Na verdade, todas as funcionalidades podem ser implementadas, sendo que, a única diferença é a passagem da sua especificação para o SharePoint Designer.

A transformação de uma Peça Web numa XSLT Data View permite principalmente alterar o seu aspecto podendo, por exemplo, criar vistas que apresentam itens em árvore agrupadas em mais do que dois níveis, ao contrário das opções no browser onde apenas é possível definir dois níveis de agrupamento de itens.

Quando se cria uma vista de uma lista, esta possibilita várias formas de filtragem e ordenação. No entanto, pelo browser não é possível dar resposta a todos os casos de filtragem e ordenação. Por exemplo, quando se tem uma lista com vários itens e se pretende mostrar os últimos seis, ordenados de forma ascendente por data de modificação, como se apresenta na figura 53, tal não é possível fazer sem recorrer ao SharePoint Designer, nomeadamente com a utilização de XSLT Filtering.






Todas Anotações			Últimas Anotações	Todas Anotações
 Início do projecto nesta data: <i>Abril 2008</i>	31-03-2008 10:19	CICERO\qualifeup		
 Primeira reunião ocorrida nesta data: <i>Abril 2008</i>	31-03-2008 10:28	CICERO\qualifeup		
 Elaboração da acta a cargo de: Eng. Silva	31-03-2008 10:34	CICERO\qualifeup		
 Revisão da acta a cargo de: Eng. Sousa	31-03-2008 10:35	CICERO\qualifeup		
 Correção da acta a cargo de: Eng. Saraiva	31-03-2008 10:37	CICERO\qualifeup		
 Aprovação da acta a cargo de: Eng. Sarmento	31-03-2008 10:43	CICERO\qualifeup		
  Acta elaborada encontra-se em ficheiro anexo	31-03-2008 10:50	CICERO\qualifeup		
 Início dos trabalhos	21-04-2008 11:36	CICERO\qualifeup		

Figura 53 – Exemplo de utilização de XSLT

Este caso pode ser resolvido, por exemplo, com a introdução de uma expressão XSLT, definida na propriedade Filtros da XSLT Data View que apresenta os itens da lista, tal como se pode observar na figura 54.

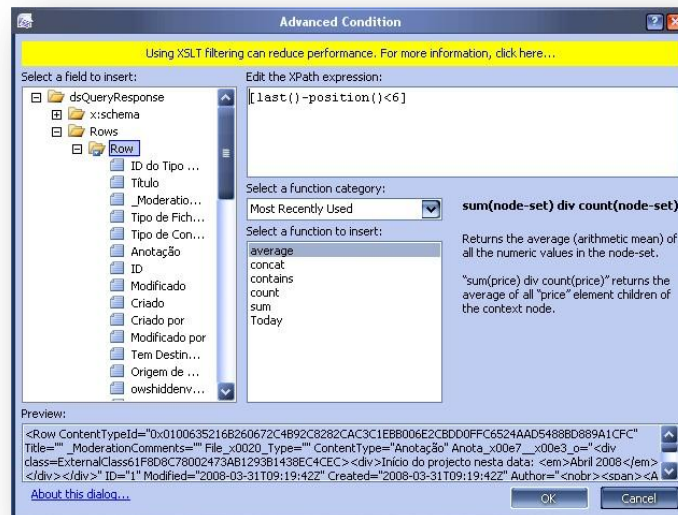


Figura 54 – Quadro de desenvolvimento de funções XSLT

Fluxos de Trabalho

Com esta ferramenta é possível também construir fluxos de trabalho mais complexos do que os criados pelo browser, através de assistente visual da própria ferramenta, ilustrado na figura 55.

Um fluxo de trabalho está sempre anexado a uma lista ou biblioteca do SP, pelo que, é necessário que o site tenha, pelo menos, uma lista ou biblioteca antes da criação de um fluxo de trabalho.

Se se pretender que o fluxo de trabalho utilize qualquer uma das colunas ou definições personalizadas, é necessário efectuar essas alterações antes da criação do fluxo de trabalho de modo a que essas colunas e definições estejam disponíveis no Estruturador de Fluxos de Trabalho.

Caso seja pretendido que o fluxo de trabalho utilize qualquer funcionalidade da lista ou biblioteca, que não esteja activada por predefinição, tal como a Aprovação do Conteúdo, é preciso activar as respectivas funcionalidades.

Relativamente à personalização pelo browser, é possível configurar uma vasta gama de opções, através definições de cada site, quer a nível de visualização, com a alteração da

página mestra, tema do site ou navegação, como já foi anteriormente referido, mas também a nível de permissões, gestão de grupos de utilizadores, criação de fluxos de trabalho, etc.



Figura 55 – Exemplo de criação de um fluxo de trabalho no SharePoint Designer

Ligações de Peças Web

Uma das opções de configuração de Peças Web passa pela possibilidade de conectar duas Peças Web, de modo a funcionarem de forma agregada, comunicando entre si com o envio de dados, permitindo impor alguma dinâmica nos conteúdos que podem ser apresentados.

A figura 56 apresenta o modo de edição de uma página que contem duas Peças Web ligadas entre si de forma a que uma delas funcione como produtora de dados e a outra como consumidora desses mesmos dados, para posterior tratamento e comportamento em função dos parâmetros recebidos.

Se, por exemplo, houver colunas comuns em duas listas, então é possível conectar duas Peças Web que sejam vistas para essas listas e ligadas pelo campo comum, de modo a que a selecção de um item provoque a actualização do conteúdo na outra Peça Web.

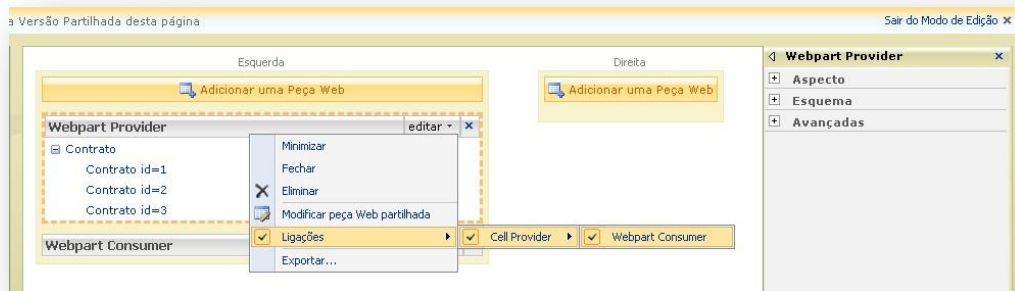


Figura 56 – Exemplo de ligações entre Peças Web

Gestão de Tipos de Ficheiros Bloqueados pelo SharePoint

O WSS possibilita a inibição de alguns tipos de ficheiros, de modo a garantir a segurança e integridade do portal. A gestão desses tipos de ficheiros pode ser feita no Centro de Administração, em *Blocked File Types*.

Personalização de Ícones no SharePoint

Para se adicionar agora um ícone ao novo tipo de ficheiro criado, é necessário editar o ficheiro DOCICON.XML, e adicionar a seguinte linha de código:

```
<Mapping Key="ext" Value="icext.gif"/>
```

Este ficheiro encontra-se em:

```
<Disco local>:\Program Files\Common Files\Microsoft Shared\Web Server  
Extensions\12\TEMPLATE\XML
```

Deve-se ter em atenção que a imagem foi adicionada à pasta IMAGES, com o nome icext.gif, onde "ext" representa a extensão do tipo de ficheiro criado. O resultado final será algo semelhante ao apresentado na figura 57.

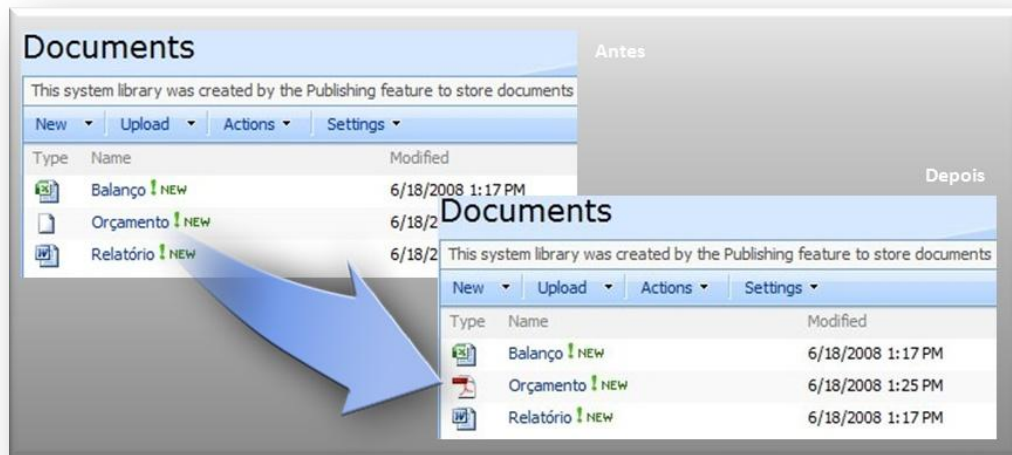


Figura 57 – Exemplo de personalização de ícones no SharePoint

Visual Studio

Tal como referido no capítulo 6, o segundo nível de personalização passa pelo desenvolvimento de código em Visual Studio. O *Object Model* é a arquitectura de programação do WSS que permite programaticamente manipular a estrutura do SP, através do acesso aos seus objectos, sendo executado do lado do servidor. Esta manipulação pode ser implementada com a linguagem C# permitindo, por exemplo:

- Adicionar, apagar e editar dados de listas
- Criar listas e bibliotecas de documentos
- Criar pastas em listas ou bibliotecas
- Criar propriedades configuráveis para peças Web
- Gestão de utilizadores
- Programação de *Event Handlers*

Exemplos de objectos que podem ser manipulados:

- Listas
- SPField
- SPList
- SPListitem
- SPView

Documentos

- SPDocumentLibrary
- SPFile
- SPFolder

Segurança

- SPGroup
- SPSite
- SPUser

Segue-se portanto, uma descrição de como desenvolver *Custom Web Parts* em VS.

Peças Web Personalizadas Desenvolvidas em Visual Studio

Para se criar uma Peça Web personalizada no VS, é preciso primeiro criar uma Class Library e adicionar as referências System.Web e Microsoft.Sharepoint.

A partir desse momento, é possível derivar a classe criada da classe Webpart e aceder aos objectos do SP com recurso ao Object Model. Para concluir a configuração da solução falta apenas criar uma Strong Name Key e acrescentar ao ficheiro assemblyinfo.cs a seguinte linha: [assembly: System.Security.AllowPartiallyTrustedCallers()].

Os passos que se seguem são para a criação da solução final compactada de modo a poder ser instalada no servidor. Para tal, é necessário criar cinco ficheiros XML que contêm informações sobre a Peça Web, os quais se encontram descritos a seguir.

Criação do manifesto da Peça Web, Class1.webpart:

```
<webParts>
  <webPart xmlns="http://schemas.microsoft.com/WebPart/v3">
    <metaData>
      <type name="ClassLibrary1.Class1, ClassLibrary1,
Version=1.0.0.0, Culture=neutral,
PublicKeyToken=xxxxxxxxxxxxxxxx" />
      <importErrorMessage>Não é possível importar esta peça
web</importErrorMessage>
    </metaData>
    <data>
      <properties>
        <property name="Title" type="string">Peça
Web</property>
        <property name="Description"
type="string">Descrição da peça web</property>
      </properties>
    </data>
  </webPart>
</webParts>
```

Criação do feature.xml:

```
<Feature Title="ClassLibrary1" Id="xxxxxxxx-xxxx-xxxx-xxxx-
xxxxxxxxxxxx" Description="" Version="1.0.0.0" Scope="Site"
Hidden="FALSE" DefaultResourceFile="core"
xmlns="http://schemas.microsoft.com/sharepoint/">
  <ElementManifests>
    <ElementManifest Location="elementmanifest.xml" />
    <ElementFile Location="class1.webpart" />
  </ElementManifests>
</Feature>
```

Criação do elementmanifest.xml:

```
<Elements xmlns="http://schemas.microsoft.com/sharepoint/">
  <Module Name="WebParts" List="113" Url="_catalogs/wp">
    <File Url="class1.webpart" Type="GhostableInLibrary" />
  </Module>
</Elements>
```

Criação do manifesto da solução, manifest.xml:

```
<Solution SolutionId="xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx"
xmlns="http://schemas.microsoft.com/sharepoint/">
  <FeatureManifests>
    <FeatureManifest Location="ClassLibrary1\feature.xml" />
  </FeatureManifests>
  <Assemblies>
    <Assembly Location="ClassLibrary1.dll"
DeploymentTarget="GlobalAssemblyCache">
      <SafeControls>
        <SafeControl Assembly="ClassLibrary1,
Version=1.0.0.0, Culture=neutral,
PublicKeyToken=xxxxxxxxxxxx" Namespace="ClassLibrary1"
TypeName="*" Safe="True" />
      </SafeControls>
    </Assembly>
  </Assemblies>
</Solution>
```

Criação do ClassLibrary1.ddf:

```
.OPTION EXPLICIT ; Generate errors
.Set CabinetNameTemplate=ClassLibrary1.cab
.set DiskDirectoryTemplate=CDROM ; All cabinets go in a single directory
```



```
.Set CompressionType=MSZIP ;** All files are compressed in cabinet files
.Set UniqueFiles="OFF"
.Set Cabinet=on
.Set DiskDirectory1=Solução

;adds manifest file
manifest.xml

;adds webpart dll
ClassLibrary1.dll

;sets the title webpart feature directory
.Set DestinationDir=ClassLibrary1

;adds the feature manifest to the feature directory
ClassLibrary1\feature.xml

;adds the element manifest to the feature directory
ClassLibrary1\elementmanifest.xml

;adds the webpart manifest to the feature directory
ClassLibrary1\Class1.webpart
```

No fim de se compilar, deve-se copiar o *assembly* gerado para a raiz da solução, criar uma pasta com o nome da solução, por exemplo ClassLibrary1, e copiar para dentro dessa pasta os ficheiros Class1.webpart, feature.xml e elementmanifest.xml.

Neste momento, o explorador da solução do VS deverá ter o aspecto da figura 58.

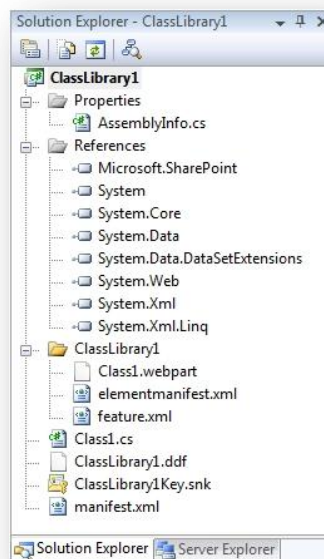


Figura 58 – Ficheiros necessários para a criação de uma Peça Web em Visual Studio

Para criar a solução compactada é necessário agora executar o comando:

```
makecab /f ClassLibrary1.ddf
```

Para finalizar a instalação no servidor basta adicionar e disponibilizar a solução através dos dois comandos seguintes:

```
stsadm -o addsolution -filename ClassLibrary1.cab
```

```
stsadm -o deploysolution -name ClassLibrary1.cab -allowgacdeployment -immediate -allcontenturls
```

Neste momento, para se utilizar a Peça Web num determinado site é preciso activá-la, nas funcionalidades da colecção de sites, e adicioná-la à página que se pretende.

A seguir apresenta-se o exemplo de uma Peça Web desenvolvida em VS.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Web;
using Microsoft.SharePoint;

namespace HelloWorld
{
    public class HelloWorldWebPart :
        System.Web.UI.WebControls.WebParts.WebPart
    {
        protected override void
            RenderContents(System.Web.UI.HtmlTextWriter writer)
        {
            writer.Write("Hello SharePoint World");

            base.RenderContents(writer);
        }
    }
}
```

Esta Peça Web produz o seguinte resultado numa página de SP, tal como se pode ver na figura 59.



Figura 59 – Exemplo de funcionamento de uma Peça Web desenvolvida em Visual Studio

Exemplo de uma Peça Web que Acede a uma Lista

Para aceder a uma lista de um site SP é necessário efectuar alguns passos, nomeadamente, abrir o site, abrir a lista, criar um SPDataSource e apresentar os dados numa SPGridView, tal como exemplificado a seguir:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Web;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using Microsoft.SharePoint;
using Microsoft.SharePoint.WebControls;

namespace AcederLista
{
    public class AcederListaWebPart :
        System.Web.UI.WebControls.WebParts.WebPart
    {
        private string __listName = "Tarefas";
        private SPGridView _spGridView;

        [
            Personalizable(PersonalizationScope.Shared),
            WebBrowsable(true),
            System.ComponentModel.Category("Opções de
            Configuração"),
            WebDisplayName("Lista")
        ]
        public string _listName
        {
```

```

        get { return __listName; }
        set { __listName = value; }
    }

    protected override void
    Render(System.Web.UI.HtmlTextWriter writer)
    {
        SPWeb _webSite =
        SPControl.GetContextSite(Context).OpenWeb();

        SPList _list = _webSite.Lists[_listName];

        SPDataSource _spDataSource = new SPDataSource();
        _spDataSource.List = _list;

        this._spGridView.DataSource = _spDataSource;
        this._spGridView.DataBind();

        base.Render(writer);
    }

    protected override void CreateChildControls()
    {
        this._spGridView = new SPGridView();
        this._spGridView.AutoGenerateColumns = false;

        BoundField _field = new BoundField();
        _field.DataField = "Titulo";
        _field.HeaderText = "Titulo";

        this._spGridView.Columns.Add(_field);
        this.Controls.Add(this._spGridView);

        base.CreateChildControls();
    }
}

```

O resultado final pode ser observado na figura 60.



Figura 60 – Exemplo de uma Peça Web personalizada que acede a uma lista

Este exemplo poderia ainda contemplar a filtragem os dados provenientes da lista através de uma *query* com recurso ao SPQuery.

De forma análoga, é possível aceder a uma biblioteca de documentos, existindo para tal outros objectos específicos, como por exemplo o SPFolder ou o SPFile.

Exemplo de um Peça Web com AJAX

Para se introduzir a tecnologia AJAX a uma Peça Web basta criar o UpdatePanel e inserir para dentro deste os componentes que se pretendem, tal como se pode ver no exemplo a seguir:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using Microsoft.SharePoint;
using Microsoft.SharePoint.WebControls;

namespace ExemploAJAX
{
    public class ExemploAJAXWebPart :
        System.Web.UI.WebControls.WebParts.WebPart
    {
        private Label _label;

        protected override void CreateChildControls()
        {
            this.EnsureUpdatePanelFixups();

            _label = new Label();

            UpdatePanel _updatePanel = new UpdatePanel();
            _updatePanel.ID = "UpdatePanell1";
            _updatePanel.ChildrenAsTriggers = true;
            _updatePanel.UpdateMode =
                UpdatePanelUpdateMode.Conditional;

            Timer _timer = new Timer();
            _timer.Interval = 1000;
            _timer.Tick += new
                EventHandler<EventArgs>(_timer_Tick);

            _updatePanel.ContentTemplateContainer.Controls.Add(
                _timer);
            _updatePanel.ContentTemplateContainer.Controls.Add(
                _label);
            this.Controls.Add(_updatePanel);

            base.CreateChildControls();
        }
    }
}
```

```

    }

    protected void _timer_Tick(object sender, EventArgs e)
    {
        _label.Text = "UpdatePanel refrescado às: " +
            DateTime.Now.ToLongTimeString();
    }

    private void EnsureUpdatePanelFixups()
    {
        if (this.Page.Form != null)
        {
            string formOnSubmitAtt =
                this.Page.Form.Attributes["onsubmit"];

            if (formOnSubmitAtt == "return
                _spFormOnSubmitWrapper();")
            {
                this.Page.Form.Attributes["onsubmit"] =
                    "_spFormOnSubmitWrapper();";
            }
        }

        ScriptManager.RegisterStartupScript(this,
            typeof(ExemploAJAXWebPart), "UpdatePanelFixup",
            "_spOriginalFormAction = document.forms[0].action;
            _spSuppressFormOnSubmitWrapper=true;", true);
    }
}
}
}

```

Este exemplo mostra as horas, actualizadas ao segundo, sem refrescar completamente a página. Tal como se pode observar, esta Peça Web não contempla o ScriptManager porque este encontra-se na página mestra, uma vez que só pode existir um por página. O resultado final pode ser observado na figura 61. De referir que, para esta Peça Web funcionar correctamente, é preciso primeiro preparar o SP para aceitar AJAX, cujo processo se encontra explicado abaixo.

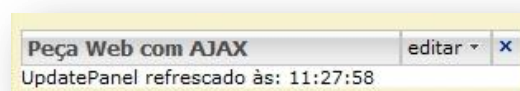


Figura 61 – Exemplo de uma Peça Web que utiliza AJAX

Este tipo de abordagem oferece uma liberdade total de personalização dos sistemas desenvolvidos, o que confere ao SP uma versatilidade na gama de soluções passíveis de implementar com esta ferramenta.

Exemplo de uma Peça Web com Ligações

Por predefinição as conexões entre Peças Web implicam um refrescamento da totalidade da página Web onde se encontram, uma vez que o SP não está preparado para trabalhar com AJAX, a menos que sejam alterados parâmetros de configuração do próprio SP.

Segue-se um exemplo da criação de duas Peças Web desenvolvidas em VS onde foi implementada uma conexão que permite que a Peça Web denominada de "Consumer" receba e apresente os valores escolhidos na peça Web "Provider".

Construção da Peça Web "Provider":

É constituída por quatro caixas de selecção que representam quatro regiões. O objectivo é passar a região seleccionada à Peça Web conectada "Consumer" sempre que o utilizador escolher uma região.



Figura 62 – Exemplo de uma Peça Web *Provider* para a ligação de Peças Web

Para trabalhar com conexões é necessário implementar interfaces em C#, pois é por este meio que os dados são transmitidos entre as Peças Web.

```
public class FilterProvider :
    System.Web.UI.WebControls.WebParts.WebPart,
    ITransformableFilterValues

private CheckBoxList _regions;
protected override void CreateChildControls ()
{
    base.CreateChildControls ();
    _regions = new CheckBoxList ();
    _regions.Items.Add (new ListItem ("North"));
    _regions.Items.Add (new ListItem ("South"));
    _regions.Items.Add (new ListItem ("West"));
    _regions.Items.Add (new ListItem ("East"));
}
```

```

        _regions.AutoPostBack = true;
        this.Controls.Add(_regions);
    }

```

Para implementar esta interface é necessário definir as seguinte propriedades:

```

public bool AllowEmptyValue
{
    get { return false; }
}
public bool AllowAllValue
{
    get { return true; }
}
public bool AllowMultipleValues
{
    get { return true; }
}
public string ParameterName
{
    get { return "Region"; }
}
public ReadOnlyCollection<string> ParameterValues
{
    get
    {
        EnsureChildControls();
        List<string> regions = new List<string>();
        for (int i = 0; i < _regions.Items.Count; i++)
        {
            if (_regions.Items[i].Selected)
            {
                regions.Add(_regions.Items[i].Value);
            }
        }
        ReadOnlyCollection<string> result = new
        ReadOnlyCollection<string>(regions);
        return result;
    }
}

```

Finalmente, basta definir o parâmetro que vai ser usado para estabelecer a conexão, neste caso o parâmetro "Region":

```

[ConnectionProvider("Region", "UniqueIDForRegionConnection",
AllowsMultipleConnections = true)]
public ITransformableFilterValues SetConnection()
{
    return this;
}

```

Este será o parâmetro que será visível na interface utilizada para a conexão, figura 63.



Figura 63 – Exemplo de configuração da ligação entre Peças Web

Construção da Peça Web "Consumer":

Esta Peça Web fará apenas a apresentação dos dados recebidos pela conexão, tendo para isso que ser criado o membro que vai guardar os dados seleccionados e inicializado o construtor.

```
public class FilterConsumer :
System.Web.UI.WebControls.WebParts.WebPart

private List<IFilterValues> _filterProviders;

private List<IFilterValues> FilterProviders
{
    get { return _filterProviders; }
}

public FilterConsumer()
{
    _filterProviders = new List<IFilterValues>();
}

protected override void Render(System.Web.UI.HtmlTextWriter
writer)
{
    foreach (IFilterValues filter in FilterProviders)
    {
        writer.WriteLine(string.Format("Parameter: {0} <br>",
filter.ParameterName));
        if (filter.ParameterValues != null)
        {
            foreach (string value in filter.ParameterValues)
            if (!string.IsNullOrEmpty(value))
                writer.WriteLine(string.Format("  value: {0} <br>",
value));
        }
    }
    base.Render(writer);
}
```

Para finalizar esta Peça Web, é necessário adicionar o método para configurar a conexão.

```
[ConnectionConsumer("filter", "UniqueIDForConsumer",
AllowsMultipleConnections = true)]
public void SetFilter(IFilterValues filterValues)
{
    if (filterValues != null)
    {
        EnsureChildControls();
        List<ConsumerParameter> parameters = new
        List<ConsumerParameter>();
        parameters.Add(new ConsumerParameter("Region",
        ConsumerParameterCapabilities.SupportsMultipleValues |
        ConsumerParameterCapabilities.SupportsAllValue));
        parameters.Add(new ConsumerParameter(
        "Status",
        ConsumerParameterCapabilities.SupportsMultipleValues |
        ConsumerParameterCapabilities.SupportsAllValue));
        filterValues.SetConsumerParameters(
        new
        System.Collections.ObjectModel.ReadOnlyCollection<Consumer
        Parameter>(parameters));
        this.FilterProviders.Add(filterValues);
    }
}
```

Após adicionar as Peças Web e configurar a conexão, basta seleccionar uma região e, automaticamente, a página é refrescada, sendo enviados os dados das regiões que se encontram seleccionadas, e apresentados na Peça Web "Consumer". O aspecto final que ilustra este comportamento está representado na figura 64.



Figura 64 – Exemplo de Peças Web personalizadas que implementam conexões

Apesar da utilidade deste tipo de funcionalidade, o facto de existir a necessidade de refrescamento de toda a página torna esta capacidade pouco prática sob o ponto de vista da experiência de utilização por parte do utilizador final. Para superar esta dificuldade, é possível construir Peças Web onde se conjuga conexões e AJAX, permitindo assim aumentar a usabilidade e performance das soluções desenvolvidas.

Para exemplificar este tipo de solução, será apresentado um exemplo em que foi criada uma Peça Web que apresenta o título de itens de uma lista, onde o objectivo é passar à outra Peça Web o identificador único (ID) de cada item, sendo refrescada apenas a zona da página que contém estas Peças Web.

A figura 65 mostra a lista que vai funcionar como fonte de dados para a Peça Web denominada por "Webpart Provider".



ID	Título
1	Contrato id=1
2	Contrato id=2
3	Contrato id=3

Figura 65 – Lista fonte de dados para a Peça Web *Provider* personalizada e com AJAX

Relativamente ao exemplo anterior, as alterações necessárias efectuar passam pela utilização de componentes AJAX que permitem o refrescamento local dos componentes contidos em Update Panels, objectos AJAX que permitem a submissão de dados ao servidor de forma parcial.

O código necessário para a construção da árvore de navegação dos itens é feita com a definição de objectos C#, *TreeView*, *TreeNode* e *RootNode*.

```
protected override void CreateChildControls()
{
    this.EnsureUpdatePanelFixups();
    updatePanell = new UpdatePanel();
    updatePanell.ID = "updatePanell1";
    updatePanell.UpdateMode = UpdatePanelUpdateMode.Conditional;

    tv = new TreeView();
    TreeNode rootnode = new TreeNode();
}
```

```

if (!this.Page.IsPostBack)
{
    tv.CollapseAll();
}

SPSite _siteCollection = SPControl.GetContextSite(Context);
SPWeb _webSite = _siteCollection.AllWebs[_siteLocation];
SPList _list = _webSite.Lists[_listName];

rootnode = new TreeNode(_list.ToString());

foreach (SPListItem _item in _list.Items)
{
    TreeNode subNode = new TreeNode(_item.Name.ToString(),
        _item.ID.ToString());
    rootnode.ChildNodes.Add(subNode);
}
tv.Nodes.Add(rootnode);

tv.SelectedNodeChanged += new
EventHandler(tv_SelectedNodeChanged);
}

void tv_SelectedNodeChanged(object sender, EventArgs e)
{
    send_data = true;
}

```

A Peça Web "Webpart Consumer" simplesmente detecta o envio de dados através da interface e, como o seu Update Panel está configurado para ser actualizado sempre que for realizado um envio de dados para o servidor, os dados enviados são automaticamente apresentados nesta Peça Web.

Aqui, ao clicarmos sobre o título "Contrato id=2" é automaticamente apresentado o ID correspondente a esse item sem refrescamento total da página.



Figura 66 – Exemplo do funcionamento de Peças Web ligadas e actualizadas com AJAX

Este comportamento permite assim a melhoria significativa da usabilidade de uma aplicação, potenciando desta forma as funcionalidades de uma ferramenta como o SP.

Event Handlers

Os eventos são mecanismos programáticos que adicionam lógica comportamental ao SP permitindo executar acções a nível do site, da lista ou dos tipos de conteúdo, para intervirem assim que detectarem algum evento, como por exemplo o adicionar um item numa lista.

Existem eventos síncronos que ocorrem antes de uma acção e eventos assíncronos que ocorrem logo após a conclusão da acção. Um possível exemplo de um evento é copiar o campo Título de um item adicionado a uma lista para outra lista, como se pode ver a seguir:

```
using System;
using System.Collections.Generic;
using System.Text;
using Microsoft.SharePoint;

namespace EventHandler
{
    public class ClassEventHandler : SPItemEventReceiver
    {
        public override void ItemAdded(SPItemEventProperties
        properties)
        {
            SPWeb _webSite =
            SPControl.GetContextSite(Context).OpenWeb();

            SPLListItem _element = properties.ListItem;

            SPLList _list = _webSite.Lists["Destino"];
            SPLListItem _listItem = _list.Items.Add();

            _listItem["Título"] = _element["Título"].ToString();
            _listItem.Update();

            base.ItemAdded(properties);
        }
    }
}
```

Depois de compilada esta solução, é necessário colocar o *assembly* gerado na *Global Assembly Cache*, GAC, do servidor e configura-lo no site pretendido, em Manage Event Handlers. De notar que, é necessário primeiro instalar a solução que fornece a gestão de eventos no SP.

AJAX no SharePoint

A Framework .NET 3.5 tem incorporada a tecnologia ASP.NET AJAX, *Asynchronous JavaScript And XML*, que permite a criação de aplicações Web dinâmicas, tendo como principal vantagem a comunicação eficiente entre a aplicação cliente e o servidor. Tendo por base funções de JavaScript, a eficiência da comunicação dos componentes AJAX deve-se ao facto de serem apenas submetidos os dados necessários, evitando o envio de toda a informação contida na página, eliminando-se assim o excesso de informação que contribui para uma pior experiência de utilização. Estamos então, desta forma, perante uma nova geração de interfaces para aplicações Web.

O SP não contempla, por predefinição, a utilização de AJAX nos seus portais. Contudo, esta indisponibilidade pode ser contornada com algumas alterações ao ficheiro de configuração de cada portal, web.config.

Dentro de `<configSections>` acrescentar:

```
<sectionGroup name="system.web.extensions"
type="System.Web.Configuration.SystemWebExtensionsSectionGroup,
System.Web.Extensions, Version=1.0.61025.0, Culture=neutral,
PublicKeyToken=31bf3856ad364e35">
  <sectionGroup name="scripting"
type="System.Web.Configuration.ScriptingSectionGroup,
System.Web.Extensions, Version=1.0.61025.0, Culture=neutral,
PublicKeyToken=31bf3856ad364e35">
    <section name="scriptResourceHandler"
type="System.Web.Configuration.ScriptingScriptResourceHandlerSect
ion, System.Web.Extensions, Version=1.0.61025.0, Culture=neutral,
PublicKeyToken=31bf3856ad364e35" requirePermission="false"
allowDefinition="MachineToApplication"/>
    <sectionGroup name="webServices"
type="System.Web.Configuration.ScriptingWebServicesSectionGroup,
System.Web.Extensions, Version=1.0.61025.0, Culture=neutral,
PublicKeyToken=31bf3856ad364e35">
      <section name="jsonSerialization"
type="System.Web.Configuration.ScriptingJsonSerializationSection,
System.Web.Extensions, Version=1.0.61025.0, Culture=neutral,
PublicKeyToken=31bf3856ad364e35" requirePermission="false"
allowDefinition="Everywhere" />
      <section name="profileService"
type="System.Web.Configuration.ScriptingProfileServiceSection,
System.Web.Extensions, Version=1.0.61025.0, Culture=neutral,
PublicKeyToken=31bf3856ad364e35" requirePermission="false"
allowDefinition="MachineToApplication" />
      <section name="authenticationService"
type="System.Web.Configuration.ScriptingAuthenticationServiceSect
ion, System.Web.Extensions, Version=1.0.61025.0, Culture=neutral,
PublicKeyToken=31bf3856ad364e35" requirePermission="false"
allowDefinition="MachineToApplication" />
    </sectionGroup>
  </sectionGroup>
```

```
</sectionGroup>
```

Dentro de `<pages>` acrescentar:

```
<controls>
  <add tagPrefix="asp" namespace="System.Web.UI"
  assembly="System.Web.Extensions, Version=1.0.61025.0,
  Culture=neutral, PublicKeyToken=31bf3856ad364e35"/>
</controls>
```

Dentro de `<assemblies>` acrescentar:

```
<add assembly="System.Web.Extensions, Version=1.0.61025.0,
Culture=neutral, PublicKeyToken=31bf3856ad364e35"/>
```

Dentro de `<httpHandlers>` acrescentar:

```
<add verb="*" path="*.asmx" validate="false"
type="System.Web.Script.Services.ScriptHandlerFactory,
System.Web.Extensions, Version=1.0.61025.0, Culture=neutral,
PublicKeyToken=31bf3856ad364e35"/>
<add verb="*" path="*_AppService.axd" validate="false"
type="System.Web.Script.Services.ScriptHandlerFactory,
System.Web.Extensions, Version=1.0.61025.0, Culture=neutral,
PublicKeyToken=31bf3856ad364e35"/>
<add verb="GET,HEAD" path="ScriptResource.axd"
type="System.Web.Handlers.ScriptResourceHandler,
System.Web.Extensions, Version=1.0.61025.0, Culture=neutral,
PublicKeyToken=31bf3856ad364e35" validate="false"/>
```

Dentro de `<httpModules>` acrescentar:

```
<add name="ScriptModule" type="System.Web.Handlers.ScriptModule,
System.Web.Extensions, Version=1.0.61025.0, Culture=neutral,
PublicKeyToken=31bf3856ad364e35"/>
```

Dentro de `<SafeControls>` acrescentar:

```
<SafeControl Assembly="System.Web.Extensions,
Version=1.0.61025.0, Culture=neutral,
PublicKeyToken=31bf3856ad364e35" Namespace="System.Web.UI"
TypeName="*" Safe="True" />
```

Dentro de <configuration> acrescentar:

```
<system.webServer>
  <validation validateIntegratedModeConfiguration="false"/>
  <modules>
    <add name="ScriptModule" preCondition="integratedMode"
type="System.Web.Handlers.ScriptModule, System.Web.Extensions,
Version=1.0.61025.0, Culture=neutral,
PublicKeyToken=31bf3856ad364e35"/>
  </modules>
  <handlers>
    <remove name="WebServiceHandlerFactory-Integrated" />
    <add name="ScriptHandlerFactory" verb="*" path="*.asmx"
preCondition="integratedMode"

type="System.Web.Script.Services.ScriptHandlerFactory,
System.Web.Extensions, Version=1.0.61025.0, Culture=neutral,
PublicKeyToken=31bf3856ad364e35"/>
    <add name="ScriptHandlerFactoryAppServices" verb="*"
path="*_AppService.axd" preCondition="integratedMode"
type="System.Web.Script.Services.ScriptHandlerFactory,
System.Web.Extensions, Version=1.0.61025.0, Culture=neutral,
PublicKeyToken=31bf3856ad364e35"/>
    <add name="ScriptResource" preCondition="integratedMode"
verb="GET,HEAD" path="ScriptResource.axd"
type="System.Web.Handlers.ScriptResourceHandler,
System.Web.Extensions, Version=1.0.61025.0, Culture=neutral,
PublicKeyToken=31bf3856ad364e35" />
  </handlers>
</system.webServer>
```

Após estas alterações ao web.config, a correcta utilização de ASP.NET AJAX com o WSS ainda não está assegurada uma vez que ambos armazenam em cache alguns tipos de acções de formulários, podendo originar alguns conflitos. Para corrigir este problema, é necessário acrescentar uma pequena função, como a que se apresenta a seguir:

```
private void EnsureUpdatePanelFixups ()
{
  if (this.Page.Form != null)
  {
    string formOnSubmitAtt =
this.Page.Form.Attributes["onsubmit"];

    if (formOnSubmitAtt == "return _spFormOnSubmitWrapper();")
    {
      this.Page.Form.Attributes["onsubmit"] =
        "_spFormOnSubmitWrapper();";
    }
  }

  ScriptManager.RegisterStartupScript(this, typeof(WebPart),
"UpdatePanelFixup", "_spOriginalFormAction =
document.forms[0].action; _spSuppressFormOnSubmitWrapper=true;",
```



```
    true);  
}
```

Por fim, é necessário ainda redireccionar a framework que se está a utilizar para a versão 3.5. Para tal, acrescenta-se ao ficheiro de configuração, dentro de `assemblyBinding`, as seguintes linhas:

```
<dependentAssembly>  
  <assemblyIdentity name="System.Web.Extensions"  
    publicKeyToken="31bf3856ad364e35"/>  
  <bindingRedirect oldVersion="1.0.0.0-1.1.0.0"  
    newVersion="3.5.0.0"/>  
</dependentAssembly>
```

CAML

A API do SP permite o acesso programático aos seus objectos através de *queries*, sendo a CAML, *Collaborative Application Markup Language*, a uma linguagem baseada em XML usada para a essa consulta de conteúdos.

Segue-se uma comparação entre a sintaxe da linguagem SQL com a CAML.

SQL:

```
WHERE ContentType = 'My Content Type'  
AND Description <> null  
AND Name = 'Anders'  
AND Surname = 'Jacobsen'  
GROUP BY Title DESC  
ORDER BY _Author
```

CAML:

```
<Where  
  <And>  
    <Eq>  
      <FieldRef Name=\"ContentType\" />  
      <Value Type=\"Text\">My Content Type</Value>  
    </Eq>  
  <And>  
    <IsNotNull>  
      <FieldRef Name=\"Description\" />  
    </IsNotNull>
```

```
<And>
  <Eq>
    <FieldRef Name=\"Name\" />
    <Value Type=\"Text\">Anders</Value>
  </Eq>
  <Eq>
    <FieldRef Name=\"Surname\" />
    <Value Type=\"Text\">Jacobsen</Value>
  </Eq>
</And>
</And>
</And>
</Where>

<GroupBy>
  <FieldRef Name=\"Title\" Ascending=\"False\" />
</GroupBy>

<OrderBy>
  <FieldRef Name=\"_Author\" />
</OrderBy>
```

LINQ para SharePoint

A Framework 3.5 disponibiliza, apenas nesta versão, um novo modelo de acesso a dados, denominada por LINQ, *Language Integrated Query*. Deste modo, é possível fazer todas as operações que o SQL permite, com a grande vantagem de não se limitar apenas a dados provenientes de uma base de dados, ou seja, é possível efectuar uma *query* a um objecto criado em C#, como por exemplo, um Array, com a grande vantagem de ser mais eficiente, comparativamente a algoritmos de ordenação e pesquisa.

Recentemente, foi lançado o LINQ para SharePoint que possibilita a utilização de *queries* para aceder a objectos do SharePoint, utilizando a sintaxe familiar do LINQ.

SmartPart

A SmartPart é uma Peça Web que aceita *User Controls* e os interpreta, apresentando-os como uma Peça Web. A vantagem de desenvolver *User Controls* consiste na possibilidade de utilização do modo de design para a construção do mesmo. Contudo, a grande desvantagem reside na falta de *IntelliSense*. Em suma, os *User Controls* são uma boa solução para quem gosta de utilizar o modo de design, enquanto que, para quem gosta apenas de código, as Peças Web oferecem uma melhor experiência de desenvolvimento.