



Licenciatura em Gestão de Sistemas e Tecnologias de Informação

## **Formulários on-line em Plataforma Colaborativa**

Elaborado por Rui Miguel Charneca

Aluno nº 2000 586



Orientador: Professora Filipa Taborda Ferreira

Barcarena

Junho 2009

Universidade Atlântica

Licenciatura em Gestão de Sistemas e Tecnologias de Informação

## **Formulários on-line em Plataforma Colaborativa**

Projecto Final de Licenciatura

Elaborado por Rui Miguel Charneca



Aluno n° 2000 00 586

Orientadora: Professora Filipa Taborda Ferreira

Barcarena

Novembro 2009





O autor é o único responsável pelas ideias expressas neste relatório



## **Agradecimentos**

À professora Filipa Taborda Ferreira pela competência com que orientou o meu trabalho, o tempo que generosamente me dedicou, ajuda e apoio em todos os momentos, e sem os quais este trabalho não teria sido possível, estou-lhe muito grato.

À minha família que me apoiou e deu sempre todo o apoio e proporcionou todas as condições para que eu pudesse atingir todos os meus objectivos.

À GG, agradeço, a paciência e esforço que teve, no encorajamento que sempre me transmitiu, sem o qual não teria conseguido terminar este trabalho.



“A Satisfação está no esforço e não apenas na realização final”

Mahatma Gandhi



## **Resumo**

### **Formulários On-line em Plataforma Colaborativa**

O presente trabalho apresenta um estudo de solução para o problema genérico da recolha de dados online.

Desenvolvido no âmbito dos projectos de investigação MUN-SI, COSI e POZ e tendo como pressuposto a futura facilidade de manutenção e escalabilidade, este estudo foca a utilização de ferramentas de Gestão de Conteúdos na obtenção, tratamento e utilização de dados em ambiente Web a partir de formulários online, disponibilizados numa plataforma Microsoft Office SharePoint com Forms Server.

Com o objectivo de facilitar a integração de uma base de dados com os dados recolhidos nos formulários exploraram-se diferentes soluções possíveis, nomeadamente para a integração do InfoPath com uma base de dados SQL chegou-se a desenvolver Web Services que servissem a comunicação entre os dois sistemas. A integração adoptada acabou por ser a mais simples e nativamente suportada pelas aplicações Office da Microsoft.

Foram ainda montados vários sites e aplicações dentro do Sharepoint, capazes de satisfazer as necessidades dos projectos.

Com vista a uma análise mais alargada no âmbito do projecto faz-se um enquadramento teórico da solução com o propósito de dar a conhecer conceitos relacionados com ferramentas colaborativas que facilmente serão parte de solução numa organização, com toda a complexidade envolvente



## **Abstract**

### **On-line Forms in a collaborative platform**

This work presents a solution study for a problem of collecting online data.

Its scope is focused in the MUNSI, COSI, and POZ investigation projects, where the objective is the future maintenance easiness and scalability of the platform, the study focuses in the use of content management tools in obtaining, treating and using data obtained in a web environment through an online form, made available by a Microsoft Office SharePoint Server using Forms Server.

With the objective of facilitating the integration of the forms collected data and a database, different possible solutions were explored, in particular, for the integration of InfoPath and SQL Server, it was necessary to develop Web Services capable to act between the two systems.

Several sites and applications were assembled in Sharepoint, capable of satisfying the needs of all the projects

In a wider review, a theoretical framework of the solution is made in order to show different concepts related to a collaborative tool, that can easily be part of a solution in an organization, with all the involving complexity.



## Índice

1	Introdução .....	23
2	Sistemas Colaborativos .....	25
2.1	Vantagens e problemas na adopção de ferramenta colaborativa .....	25
2.2	Sistemas colaborativos CMS .....	27
2.2.1	Capacidades do sistema .....	27
2.3	Sistemas colaborativos WCM.....	27
2.3.1	Capacidades do Sistema.....	28
2.4	CMS vs WCM .....	29
2.4.1	WCM Open Source.....	29
2.4.2	WCM Comercial.....	29
2.4.3	Vantagens e desvantagens .....	30
2.5	Considerações na adopção de solução .....	32
3	Plataforma Centralizada de suporte ao projecto .....	34
3.1	Vantagens da utilização da plataforma MOSS .....	35
4	Enquadramento da solução .....	36
4.1	Situação na organização.....	36
4.2	Requisitos funcionais da solução a implementar .....	37
4.3	Proposta de solução .....	38
5	Abordagem à solução.....	40
5.1	Criação e parametrização da Plataforma Sharepoint .....	40
5.1.1	Web Application Base .....	43
5.1.2	Site Collection Base.....	43
5.1.3	Internet Web Application.....	44

5.1.4	Web.config.....	45
5.1.5	Activação de Forms Authentication .....	47
5.1.6	Sites dos grupos de Investigação .....	48
5.2	Autenticação .....	50
5.2.1	Autenticação FBA e SQL .....	52
5.2.2	Criação da Base de dados FBAASPNetDB .....	52
5.2.3	Aplicação Web de gestão de users.....	53
5.3	O InfoPath na criação de formulários .....	55
5.3.1	Business Logic do Formulário.....	57
5.4	Soluções para a integração de dados.....	59
5.4.1	O armazenamento de dados em Listas Sharepoint .....	60
5.4.2	A integração nativa com Listas de Sharepoint.....	64
5.4.3	Utilizar Web Services na integração de dados.....	65
5.4.4	Utilizar Workflows na integração de dados.....	73
5.5	O Interface no processo de recolha de dados e partilha de informação.....	73
5.5.1	Recolha de dados – Interface Utilizador.....	74
5.5.2	Análise dos dados recolhidos – Interface Administração .....	76
6	Trabalhos Futuros .....	81
7	Conclusão.....	83
8	Bibliografia .....	85
Anexo A.	Arquitectura Sharepoint.....	1
A.1	Arquitectura WSS .....	1
A.2	Arquitectura MOSS .....	2
A.2.1	Arquitectura Lógica .....	4

A.2.2	Recursos Necessários.....	5
A.2.3	Arquitectura Física.....	6
A.2.4	Arquitectura administrativa .....	10
A.2.5	Segurança no Sharepoint .....	10
A.2.6	Anonymous Authentication .....	13
A.2.7	Windows Authentication .....	13
A.2.8	Forms Based Authentication.....	13
A.2.9	Permissões Sharepoint .....	14
A.2.10	Hierarquia e Herança .....	16
A.2.11	Utilizadores e grupos de Sharepoint .....	17
A.2.12	Tipos de conteúdos e funcionalidades .....	17
A.2.13	Site Content Type .....	18
A.2.14	Bibliotecas e Listas de Sharepoint .....	18
A.2.15	Web Parts .....	19
A.2.16	Workflows .....	20
Anexo B.	Web Service .....	21
B.1	SOAP .....	22
Anexo C.	Form Server e Microsoft Office InfoPath 2007 .....	23
C.1	Microsoft Office InfoPath 2007 .....	23
C.2	Integração InfoPath e SQL Server .....	27
C.3	Forms Services.....	29
C.3.1	Opção Form Services para Web browser .....	29
C.3.2	Requisitos Mínimos para o Form Services .....	30
C.3.3	Browsers Suportados .....	31

Anexo D.	Plataforma a instalar .....	32
Anexo E.	Código e Scripts Utilizados .....	34
E.1	Script SQL .....	34
E.1.1	Script SQL da Base de Dados .....	34
E.1.2	Script SQL da Tabela Inquerito .....	36
E.1.3	Código-fonte C# do Web Service .....	37
Anexo F.	Manual de utilizador .....	41
Anexo G.	Visio da Estrutura Implementada .....	42

## Índice de figuras

Figura 1 – Estrutura do site de Administração Central Sharepoint .....	42
Figura 2 – Portal Inicial .....	44
Figura 3 - Web.config Web application base e extended .....	46
Figura 4 – Web.config administração.....	46
Figura 5 - Formulário de início de sessão.....	48
Figura 6 – Estrutura da Plataforma instalada.....	50
Figura 7 – Página de entrada sem autenticação .....	51
Figura 8 – Diagrama da Base de Dados de autenticação .....	53
Figura 9 – Web.config da Web application de autenticação .....	54
Figura 10 – Configuração Provider e criação de utilizador .....	55
Figura 11 - Layout do formulário .....	56
Figura 12 Vistas de validação .....	58
Figura 13 – Business Logic do Formulário .....	59
Figura 14 - Vista Biblioteca de Documentos.....	60
Figura 15 - Bibliotecas de Formulário.....	62
Figura 16 - Acesso á Home page .....	74
Figura 17 - Vista de biblioteca de formulários .....	75
Figura 18 – Formulário no browser .....	75
Figura 19 - Vista de funcionalidades .....	77
Figura 20 – Lista Exportada para Excel.....	78
Figura 21 - Vista Tabela em Access .....	79
Figura 22 - Evolução da Arquitectura Sharepoint .....	1
Figura 23 – Arquitectura WSS.....	2

Figura 24 – Arquitectura MOSS 2007 .....	3
Figura 25 - Pilares do MOSS .....	4
Figura 26 - Estrutura Single Server .....	7
Figura 27 Estrutura Small Farm.....	8
Figura 28 - Estrutura Média Server Farm.....	8
Figura 29 - Estrutura Large Server Farm.....	9
Figura 30 – Topologia e performance.....	9
Figura 31 – Zonas e Métodos de autenticação do Sharepoint .....	12
Figura 32 – Lista de permissões no Sharepoint .....	14
Figura 33 – Níveis de permissões .....	15
Figura 34 – Securable Objects .....	16
Figura 35 – Relações de herança .....	16
Figura 36 – Utilizadores e Grupos no Sharepoint.....	17
Figura 37 – Comunicação Web Services .....	21
Figura 38 - Comunicação InfoPath Web Service .....	22
Figura 39 - Estrutura SOAP .....	22
Figura 40 - Restrições InfoPath .....	28
Figura 41 – Arquitectura a instalar .....	32
Figura 42 - Script de criação da Base de Dados .....	35
Figura 43 - Script de criação da Tabela .....	36
Figura 44 - Código C# do Web Service.....	40

## Índice de tabelas

Tabela 1 – MembershipProviders Intranet e Internet .....	47
Tabela 2 - Estrutura Sites de Inquérito .....	49
Tabela 3 - Características de field em InfoPath .....	56
Figura 4 - DataSource do formulário .....	57
Tabela 5 - Opções do formulário .....	63
Tabela 6 - Características de Main submit.....	63
Tabela 7 - Características da tabela da base de dados Inquerito.....	65
Tabela 8 - Características IIS do Web Service .....	67
Tabela 9 - Características de Sharepoint Library Submit .....	67
Tabela 10 - Características de WebService Submit.....	68
Tabela 11 – Passagem de parâmetros para o web service: método addNewInq.....	68
Tabela 12 - Características de Sharepoint Library Update .....	69
Tabela 13 - Características do Web Service Update.....	70
Tabela 14 - Passagem de parâmetros para o web service Update: método updateInq ...	70
Tabela 15 - Características de Sharepoint Library Delete .....	71
Tabela 16 - Características do Web Service Update.....	72
Tabela 17 - Passagem de parâmetros para o web service Update: método updateInq ...	72
Tabela 18 - Exemplo de web query para Excel .....	78
Tabela 19- Tabela de Requisitos.....	5
Tabela 20 – Tabela de browsers compatíveis .....	31
Tabela 21 - Métodos do Web Service.....	37



## **Lista de Abreviaturas e Siglas**

<b>AD</b>	Active Directory
<b>API</b>	Application Programming Interface
<b>ERP</b>	Enterprise Resource Planning
<b>FBA</b>	Forms Based Authentication
<b>HTML</b>	Hypertext Markup Language
<b>HTTP</b>	Hypertext Transfer Protocol
<b>IIS</b>	Internet Information Service
<b>IT</b>	Information Technologies
<b>LDAP</b>	Lightweight Directory Access Protocol
<b>MOSS</b>	Microsoft Office Sharepoint Server
<b>MSDN</b>	Microsoft Developer Network
<b>OU</b>	Organizational Unit
<b>PHP</b>	Hypertext Preprocessor
<b>SLA</b>	Service Level Agreement
<b>SMTP</b>	Simple Mail Transfer Protocol
<b>SOAP</b>	Simple Object Access Protocol
<b>SPS</b>	Sharepoint Portal Server
<b>SQL</b>	Structured Query Language
<b>SQL</b>	Structured Query Language

<b>SSO</b>	Single Sign On
<b>STS</b>	Sharepoint Team Services
<b>UDC</b>	Universal Data Connection
<b>URL</b>	Uniform Resource Locator
<b>WCM</b>	Web Content Management
<b>WFE</b>	Web Front End
<b>WSS</b>	Windows Sharepoint Services
<b>XML</b>	Extensible Markup Language

## **1 Introdução**

Numa era em que a informação assume mais do que nunca um papel activo e determinante na construção do conhecimento, torna-se cada vez mais necessário inovar e desenvolver a eficácia dos sistemas de pesquisa e de processamento da informação. Na chamada Sociedade da Informação ou do Conhecimento, mais do que nunca, a informação e a colaboração entre pares é vital para o processo de tomada de decisão da organização.

A questão central deste trabalho foca-se na forma de recolher dados em tempo real recorrendo aos Sistemas de Informação.

O papel dos Sistemas de Informação num projecto de investigação e a sua inserção na estratégia de recolha e análise de dados é da maior importância, pois permite que seja possível recolher, agregar e analisar grandes conjuntos de dados, a partir de diferentes ferramentas e fontes, possibilitando que a investigação atinja os objectivos propostos sustentada numa estrutura de dados e conhecimento válidos e fiáveis.

Todo um processo de recolha de dados num ambiente, como por exemplo um ambiente Web, permite uma eficaz abrangência com menores custos humanos e materiais quando comparado com a recolha de dados tradicional via questionário impresso.

Neste trabalho propõe-se a utilização de um sistema de trabalho colaborativo, que possibilite a criação, publicação, organização e análise de dados recolhidos em tempo real em formulários, acessíveis a todos os participantes no projecto de investigação,

Reflectindo-se sobre a solução do problema principal que constitui o objectivo deste trabalho concluiu-se que uma ferramenta colaborativa resolvia grande parte do problema pelas suas características nomeadamente a da publicação e partilha de informação sobre resultados e pressupostos. Também a comunicação entre os investigadores envolvidos foi considerada uma mais-valia assim como a organização e sectorização da informação dos diferentes projectos do centro de investigação. As razões indicadas puseram de parte, logo à partida, o desenvolvimento de software à medida.

Escolhida a plataforma da Microsoft como suporte colaborativo de interface Web identificaram-se três questões que constituíram os objectivos parciais deste projecto uma

vez que o próprio Sharepoint resolve grande parte do desenvolvimento necessário a uma aplicação Web para recolha e partilha de dados:

Primeiro – A questão da “Autenticação” uma vez que a solução não estaria integrada com a Active Directory do domínio.

Segundo – A criação do interface que passou pela parametrização da plataforma e criação das aplicações.

Terceiro – O Estudo de solução para resolver a questão da integração dos dados recolhidos com uma Base de Dados para o tratamento dos mesmos.

Neste texto, faz-se inicialmente uma descrição introdutória às tecnologias e plataformas colaborativas, ao que se segue a justificação da escolha da plataforma. Apresentam-se então os pressupostos que existem no âmbito do projecto, limitações e requisitos principais que levaram aos objectivos parciais do trabalho que constituíram o estudo de solução. Apresenta-se a forma como se solucionou cada um dos problemas e respectivas hipóteses de solução, referindo as possibilidades, forças e fraquezas e por fim tecem-se algumas considerações como conclusão.

## **2 Sistemas Colaborativos**

Muitas organizações, com os seus sites ou portais, sofrem de um grave problema de controlo de informação. Por razões de crescimento, de desenvolvimento tecnológico, ou por necessidades específicas, utilizam no seu dia-a-dia, dados provenientes de “ilhas” de informação, cada uma com os seus sites, formatos, recursos, etc. tornando-os fechados, e envolvendo processos de trabalho morosos e difíceis.

Descrevem-se as vantagens que advêm da adopção de um ambiente único e colaborativo e também os principais problemas que levam à adopção de uma filosofia e infra-estrutura colaborativa. Identificam-se/descrevem-se os principais tipos de ferramentas colaborativas.

### **2.1 Vantagens e problemas na adopção de ferramenta colaborativa**

Podemos imaginar o impacto numa organização em que é possível ser tomada uma decisão chave em que não são contemplados todos os dados e variáveis possíveis. As dificuldades aumentam se existir necessidade de juntar informação de fontes distintas, em que é necessário desenvolver ferramentas de agregação e análise dos dados, transformá-los e torná-los legíveis para o utilizador. Além disso, na perspectiva do utilizador final, existe o problema a utilização de ferramentas complexas. O utilizador quer utilizar ferramentas familiares e que conhece e utiliza no dia-a-dia, como o Outlook ou um browser.

Há assim uma necessidade de agregar a informação vinda dessas fontes e torná-la acessível a toda a organização, de modo a permitir que:

- a) Os utilizadores possam tomar decisões informadas e rápidas, a partir de dados transversais á organização.
- b) Os administradores de IT devem poder administrar e gerir os dados para toda a organização com facilidade e de modo transparente.
- c) O desenvolvimento deve ser feito numa plataforma comum, com serviços e ferramentas que permitam o trabalho a partir de diferentes fontes
- d) Os utilizadores finais devem poder utilizar as ferramentas do seu dia-a-dia para fazer o seu trabalho habitual.

Segundo Kathy Harris, identificaram-se algumas razões que poderão ser responsáveis pela referida dificuldade de integração e a criação das denominadas “ilhas”.

Os principais entraves são:

- a) Bases de dados corporativas, como por exemplo sistemas SAP ou CRM Siebel, necessitam de recursos muito especializados no desenvolvimento e utilização.
- b) Aplicações corporativas desenvolvidas a pedido, que também necessitam de recursos especializados.
- c) Processo de publicação cada vez mais complexo e exigente.
- d) Pouca adesão a soluções WCM.
- e) Pouca integração entre sistemas internos e sistemas WCM.
- f) Métodos de trabalho enraizados e mentalidades dos utilizadores.

A realidade é que as soluções “*top-down*”, que permitem uma gestão e administração centralizada, são pouco flexíveis e funcionais. São totalmente dependentes do fabricante e pouco abertas a interacção de fontes abertas ou standard.

Soluções isoladas podem servir um fim imediato, mas criam “ilhas de informação” que estão desligadas da organização global. Este isolamento não trará benefícios numa perspectiva de uma organização com sistemas de informação. Uma grande percentagem da informação corporativa está nos discos e pastas pessoais dos indivíduos, cujo acesso é limitado. Estudos recentes indicam que 50 a 70% da informação relevante a uma organização tem origem em fontes de outros indivíduos.

Com a diversidade de fontes e tipos de dados disponíveis como imagens e texto, é cada vez mais complexo o processo de publicação de conteúdos. Ao mesmo tempo, há cada vez mais pressão para esse trabalho ser realizado de modo cada vez mais rápido.

Os WCM eram normalmente difíceis de utilizar. As curvas de aprendizagem dos autores de conteúdo eram geralmente longas e pouco integradas com as ferramentas já utilizadas. A falta de integração entre sistemas internos e sistemas Web leva a que sejam gastos recursos e se dupliquem os sistemas de conteúdos. O que pode ainda provocar que um deles seja

menosprezado em detrimento do outro, levando a que o conteúdo Web seja desactualizado ou com dados pouco fiáveis

## **2.2 Sistemas colaborativos CMS**

O CMS (*Content Management System*) é uma plataforma que através de um conjunto de regras, processos e workflows permitem criar, editar, rever, indexar, procurar publicar e arquivar vários tipos de dados e informação. É normalmente utilizado numa organização para controlo de documentos específicos á sua actividade, como por exemplo, manuais, brochuras, tabelas de resultados ou preços, etc., e pode utilizar vários tipos de dados como imagens, áudio, vídeo, documentos, etc. Deste modo, permite gerir informação e conteúdos, independentemente do formato ou local, mas com regras e processos pré definidos que garantem que os conteúdos são coerentes e válidos.

### **2.2.1 Capacidades do sistema**

- a) Com um CMS é possível identificar todos os utilizadores intervenientes, definir os seus papéis e responsabilidades.
- b) Pode ser definido um workflow colaborativo, por exemplo, definindo a responsabilidade de aprovação de um documento se existirem alterações que necessitem de ser revistas e aprovadas por um superior.
- c) Possibilita rastrear e gerir várias versões do mesmo conteúdo.
- d) Pode ainda publicar conteúdo para um repositório que disponibiliza o acesso a esse mesmo conteúdo, com propriedades de pesquisa organizativa e obtenção de conteúdos.

## **2.3 Sistemas colaborativos WCM**

A denominação WCM (*Web Content Management*) é uma extensão de CMS, onde possibilita e simplifica a publicação de conteúdos para sites Web, permitindo que os autores submetam esses conteúdos de um modo fácil, através de modelos/templates predefinidos,

onde o utilizador pode criar os seus conteúdos sem ter conhecimentos técnicos de programação ou desenvolvimento Web.

Foca principalmente a criação e edição de páginas, facilitando o controlo de conteúdos a um nível pouco técnico e de fácil compreensão para os utilizadores.

### **2.3.1 Capacidades do Sistema**

- a) O WCM permite que a informação seja actual, consistente, disponível e com qualidade.
- b) Permite a reutilização de conteúdos, quer seja através de plataformas ou aplicações diferentes, reflectindo os benefícios na produtividade.
- c) Por ser um serviço Web, permite acesso a novas audiências, através de qualquer dispositivo com acesso á internet.
- d) A criação de conteúdos é descentralizada, onde a contribuição é global, e a disponibilidade de conteúdos é extremamente rápida, em comparação com qualquer outro tipo de sistema
- e) Os criadores dos conteúdos podem assumir a responsabilidade da informação que disponibilizam.
- f) Facilita o processo centralizado de workflows, aprovação regras e processos, permitindo a criação sem perda de controlo sobre os conteúdos, garantindo que os processos internos são seguidos e onde toda a criação é passível de rastrear e responsabilizar.
- g) O WCM oferece uma vantagem competitiva ou elimina uma desvantagem competitiva, pois a página Web é cada vez mais a “cara” da empresa, e um serviço Web sempre actualizado e dinâmico dá a ideia de uma organização evoluída tecnologicamente.
- h) Permite uma resposta muito rápida a alterações num ambiente muito competitivo.

## **2.4 CMS vs WCM**

Numa organização, este é um problema de grande complexidade, que exige uma análise cuidada e o encontrar uma solução deverá ser uma medida ponderada, pois existem várias ferramentas disponíveis no mercado, com as suas vantagens e limitações. A considerar, desde sistemas proprietários como o Sharepoint da Microsoft ou o Documentum da EMC a *Open Source* como o Drupal ou o Alfresco, sendo equivalentes em alguns aspectos, mas diferem em funcionalidades, tecnologia, preços e suporte. Muitos dos CMS disponíveis, permitem a publicação de conteúdos básicos, mas a sua função principal é a de Gestão de conteúdos estáticos e documentos. Um WCM deverá permitir uma navegação não-linear sem perder a capacidade de ter um controlo granular de conteúdos individuais, deste modo fornecendo um serviço dinâmico, flexível e escalável fugindo á complexidade dos CMS.

### **2.4.1 WCM Open Source**

Segundo Stephen Powers et al., várias organizações começam cada vez mais a analisar a opção de um WCM *Open Source* como o Drupal ou o Alfresco, devido ao facto da insatisfação com WCMs comerciais, orientados aos seus produtos complementares, com custos elevados, e com pouco impacto na organização os levar a procurar soluções independentes. As soluções *Open Source* escaláveis e são desenvolvidas em *open standards*, arquitectura Java ou PHP.

### **2.4.2 WCM Comercial**

O Microsoft Sharepoint Server ou o EMC Documentum, são sistemas integrados de funcionalidades que permitem que uma organização possa com facilidade gerir conteúdos de forma centralizada, acelerando os seus processos numa perspectiva colaborativa e transversal, permitindo aos profissionais de IT, através de ferramentas, de uma plataforma que permite que seja administrável, extensível e transparente.

### 2.4.3 Vantagens e desvantagens

As vantagens e desvantagens entre uma solução comercial e uma solução *Open Source* são diversas pois não existe nenhuma solução perfeita para todos os problemas. Apesar do custo imediato poder ser um factor de decisão, este deve ser considerado mas não deve ser o único factor a ter em conta.

As vantagens do *Open Source* sobre soluções comerciais são:

- a) Custos: Os custos podem ser praticamente nulos numa solução *Open Source* relativamente aos custos das licenças do software. Uma solução comercial tem custos directos nas licenças de software adquirido, além de todos os recursos envolventes dos quais depende como o Sistema Operativo ou Bases de dados e em alguns casos as licenças necessitam ser renovadas anualmente.
- b) Segurança: Numa solução *Open Source* todo o código fonte está disponível e pode ser utilizado, alterado e corrigido. As soluções comerciais são “*closed source*”, onde o código fonte não está disponível, o que pode implicar que possíveis problemas não sejam detectados antecipadamente e corrigidos.
- c) Estabilidade e Performance: Sistemas *Open Source* têm uma base de possibilidades de instalação mais abrangente, testada e otimizada para diferentes soluções. Apesar de tudo, podem não fornecer totalmente, a performance, segurança, fiabilidade e funcionalidades necessárias para atingir o objectivo, que é um serviço de alta qualidade. No caso das soluções comerciais, o foco é direccionado especificamente para as suas soluções, o que podendo ser limitativo, permite que situações inesperadas ou imprevistas sejam mais limitadas.
- d) Suporte: Na maioria dos casos, o suporte para soluções *Open Source* é pago, e muitas vezes a preços superiores aos das soluções comerciais, mas também são disponibilizadas abertamente outras fontes como fóruns e *mailing-lists*. A documentação pode ser muito pouca ou pouco actualizada e dirigida a um público mais técnico e não para o utilizador final. Nos

sistemas comerciais o suporte é pago, e na maioria, fontes como fóruns são praticamente inexistentes mas a documentação é abundante.

Contudo, há vantagens claras das soluções comerciais que as soluções *Open Source* não podem oferecer:

- a) **Segurança no Futuro:** Numa solução comercial, a continuidade do desenvolvimento pode ser garantido contratualmente por períodos de tempo. O desenvolvimento do sistema *Open Source* pode acabar abruptamente.
- b) **Know-how e desenvolvimento:** O fabricante possui um extenso conhecimento e *know-how* dos seus produtos. O *Open Source* pode não ter pessoas com as *skills* pretendidas no conjunto global, pois muitas vezes é dependente da boa vontade de quem contribui.
- c) **Estabilidade e performance:** Uma solução comercial garante que as pessoas certas fazem o trabalho certo. Soluções *Open Source* podem ser desenvolvidas por pessoas com poucas capacidades de desenvolvimento.
- d) **Garantia:** Existe garantia contratual numa solução Comercial. Numa solução *Open Source* isto já não acontece.
- e) **Conceptualização:** Num sistema comercial, o planeamento do projecto de desenvolvimento é feito por profissionais. Num sistema *Open Source* isto pode não acontecer.

Temos vindo a assistir a uma evolução na abordagem, com uma visão cada vez mais abrangente e integradora em pontos que eram anteriormente relevados para um segundo plano, passando por processos cada vez mais complexos e recorrendo a diferentes tipos de requisitos e inputs, não apenas técnicos e sim numa visão global da organização como um todo sem perder o foco nos objectivos finais projectados.

Em resumo, ambos os tipos de soluções devem ser considerados e integrados numa selecção cuidada e profissional de escolha de um WCMS, onde as Soluções *Open Source* devem tentar suplantar as soluções comerciais, do mesmo modo que as soluções comerciais se tentam suplantar entre elas.

## 2.5 Considerações na adopção de solução

Para evitar uma escolha errada sobre qual o caminho a escolher, seja qual for a opção, deve ser utilizada uma metodologia de análise e planeamento de modo a que o investimento de recursos sejam um motor de crescimento e mudança com base nas necessidades e recursos disponíveis.

De um ponto de vista das TI, as considerações recaem sobre a infra-estrutura subjacente, as ferramentas de desenvolvimento e capacidades, conforme os seguintes pontos:

- a) Facilidade de utilização: A criação e edição de conteúdos deverão ser fácil por parte de utilizadores com poucos conhecimentos técnicos.
- b) Controlo: O controlo do WCM deve ser sempre da organização, mesmo que envolva departamentos não técnicos.
- c) Integração: As aplicações organizacionais devem ser apoiadas pela experiência e pelo suporte. Funcionalidades de pesquisa, monitorização, análise e segmentação podem otimizar a experiência do utilizador.
- d) Flexibilidade: De modo a fornecer uma experiência mais dinâmica ao utilizador, deve permitir várias linguagens e ser possível de utilizar vários suportes sem perda de qualidade dos conteúdos.
- e) Arquitectura: O WCM deve ter uma visão de conteúdo estruturado e separar o conteúdo da apresentação, criando dinamicamente cada página juntando cada bloco de informação para essa mesma página, e permitindo que no seu conjunto seja possível aceder aos conteúdos de várias formas.
- f) Desenvolvimento: O WCM deve permitir com facilidade a criação, manutenção e desenvolvimento de ferramentas, controlos e capacidades.
- g) Suporte integrado de:
  - i. Pesquisa
  - ii. Acessibilidade.

- iii. Segurança: possibilitando gestão de permissões e autenticação externa e interna sobre sistemas de autenticação organizativos.
  - iv. Múltiplos sites com diferentes linguagens e adaptados á experiência e recursos do utilizador
  - v. Navegação, funções automatizadas que permitem a criação e edição de novas páginas, secções ou sites.
- h) Escalabilidade e Performance: Deve ser considerado o *overhead* de manutenção e custo total de *ownership*. Também é importante o suporte de produção de múltiplos sites num único sistema.

Assim, de um ponto de vista da organização, tanto para os utilizadores finais como para os decisores, não deve ser apenas uma escolha de Sistema ou tecnologia, mas também como um modo de atingir objectivos reais da organização.

### **3 Plataforma Centralizada de suporte ao projecto**

Para a recolha de dados online e partilha de informação, objectivo do projecto onde este trabalho tem cabimento, utilizou-se um modelo centralizado de informação, pode-se assim dar mais garantias de que os dados recolhidos são uniformes em toda a sua extensão.

O Microsoft Office SharePoint Server foi a plataforma seleccionada pois é a opção que nos oferece mais garantias de funcionalidade, estabilidade e integração com todos os sistemas já existentes, todos de base Microsoft, desde o sistema operativo até às aplicações a nível de utilizador como o Microsoft Office.

Utilizar uma ferramenta que não seja completamente “estranha ao utilizador” é um ponto importante, onde apesar da interface Web, todas as funcionalidades são próximas de aplicações como o Microsoft Outlook ou o Microsoft Excel.

Através de uma plataforma como o Microsoft Office SharePoint Server, podemos ter através de um interface Web, um ambiente seguro, com autenticação e permissões facilmente configuráveis a utilizadores ou grupos, garantindo a segurança e que cada utilizador tenha as permissões necessárias para realizar o seu trabalho.

Relativamente às questões de licenciamento, que no caso desta plataforma é sem dúvida de um valor bastante considerável, foi ultrapassado visto que existem na organização protocolos que garantem um licenciamento gratuito ao abrigo do acordo msdnAcademicAliance da Universidade participante no projecto.

Também o suporte e documentação são acessíveis e abundantes tanto em literatura como fóruns.

A plataforma Sharepoint, baseada na Framework .NET, permite uma melhor extensibilidade e customização, seja na integração com outros sistemas como na criação de Web Parts com o Visual Studio usando linguagens C# ou VB.NET.

A administração é centralizada e pode ser realizada pela organização, sendo independente de terceiros.

O Sharepoint Forms Server dá suporte à funcionalidade web de suporte aos formulários independentemente da instalação do InfoPath, ferramenta utilizada para os criar.

A plataforma Microsoft Office Sharepoint Server 2007 tem a possibilidade adicional de permitir criar bibliotecas de documentos e calendários partilhados, Web Parts, gerir permissões de grupo, entre outras mais-valias resultantes da utilização da plataforma no âmbito do trabalho colaborativo.

### **3.1 Vantagens da utilização da plataforma MOSS**

Um aspecto fundamental na utilização de uma plataforma deste nível é sem dúvida a facilidade de utilização, aspecto dinâmico e corporativo sendo familiar ao utilizador, que pode, através de alguns passos adicionar ou criar funcionalidades, disponibilizando-as a utilizadores ou grupos de utilizadores específicos livremente, e tudo isto sem que seja necessário qualquer tipo de conhecimentos de programação ou desenvolvimento. Deste modo, o que antes necessitava da intervenção de técnicos especializados em desenvolvimento, pode ser agora feito por qualquer colaborador da organização, numa plataforma centralizada e disponível a todos. Essa informação pode ser disponibilizada em vários tipos de formato, sem comprometer a sua integridade e acedida através de ferramentas dinâmicas, apelativas e familiares a um utilizador. A curva de aprendizagem é praticamente nula.

Todos os problemas relativos a sistemas próprios desenvolvidos á medida, que teriam de ser alterados, personalizados ou refeitos deixam de existir, apesar de pontualmente algumas opções terem de ser desenvolvidas a pedido, a gestão da plataforma pode ser praticamente autónoma, ou pelo menos gerida internamente, sem recorrer a equipas de desenvolvimento.

Assim, o propósito da plataforma, é disponibilizar serviços, com foco em quem produz a informação e de quem a usa, de um modo totalmente dependente de quem a cria, num modelo de convergência a todos os níveis, seja a plataforma, os dados, as ferramentas, a análise ou a decisão. Assim, minimizam-se custos de comunicação, disparidades e erros nos dados em análise, com ganhos efectivos em tempo simplificação e facilidade de utilização dos processos.

## **4 Enquadramento da solução**

Iremos aqui referir relativamente á organização, o que existe, os problemas encontrados e possíveis soluções no enquadramento do problema e solução a desenvolver.

### **4.1 Situação na organização**

No departamento de investigação, berço dos projectos MUN-SI, POZ e COSI, são efectuadas regularmente trabalhos de recolha de dados em inquéritos.

Nos trabalhos realizados pelo centro, os dados referentes aos diferentes estudos são recolhidos por contacto directo ou em papel, através de formulários impressos, que são preenchidos pelos próprios investigadores ou pelos inquiridos, dependendo dos requisitos da investigação em causa.

Apesar da orientação dos investigadores, têm sido encontradas falhas no preenchimento, a nível de erros, ou omissão de dados por parte dos inquiridos. Esta situação terá certamente impacto na qualidade dos dados recolhidos

Os inquéritos, depois de reunidos, são todos processados a partir do papel para um ficheiro Excel previamente estruturado ou apresentados a sistemas de leitura óptica, sendo que todo este processo é inteiramente manual, moroso e propenso a erros que podem advir da falta de organização dos documentos, falha humana na introdução dos dados, falta de controlo de erros, falta de procedimentos correctivos ao encontrar um erro ou falta de dados, tipos de dados incorrectos, entre outras situações adversas.

Numa análise do ficheiro Excel, depressa constatamos que o próprio preenchimento é difícil e nada intuitivo. Nas tabelas, a informação é escassa, densa e codificada o que dificulta a introdução manual dos dados.

Este ficheiro Excel, depois de preenchido na totalidade é importado para uma base de dados do SPSS<sup>1</sup> para ser processado e analisado.

## 4.2 Requisitos funcionais da solução a implementar

- Acesso aos formulários realizado por Web browser a partir de qualquer lugar.
- Acesso á plataforma autenticado pela Web para utilizadores e localmente para administradores.
- Gestão de permissões. Nenhum utilizador poderá, mesmo que autenticado, aceder ao Site de outro grupo de investigação.
- Um utilizador não autenticado poderá apenas aceder à informação pública dos projectos e nunca aos formulários.
- Permissões definidas por grupos de utilizadores, tanto no acesso a inquéritos como nas operações a realizar.
- Acesso simples e intuitivo aos formulários.
- O acesso, preenchimento e envio de formulários é totalmente independente de o cliente ter ou não a aplicação InfoPath instalada no seu Sistema.
- O formulário será graficamente apelativo e dinâmico, sem descurar a funcionalidade de recolha de dados.
- Os formulários serão preenchidos numa estrutura semelhante á dos formulários em papel.
- O formulário terá funcionalidades de controlo de dados e erros, notificando o utilizador em caso de erro ou falha antes do envio.
- Apenas um formulário correctamente preenchido e sem erros poderá ser submetido á plataforma.
- Qualquer actualização no formulário base estará imediatamente disponível e dentro do possível, sem impacto nos dados já submetidos.
- A recolha de dados será sempre que possível em tempo real. Deve existir a possibilidade de envio de formulários únicos ou em bloco no caso de não existir acesso Web no local do cliente.

---

<sup>1</sup> SPSS - *Statistical Package for the Social Sciences* – Aplicação de análise estatística

- Possibilidade de correcção de um formulário no browser, mesmo depois de submetido para a plataforma.
- A criação de sites e formulários deve ser facilitada ao máximo e dentro do possível sem intervenção a nível de desenvolvimento específico futuro de modo a garantir a liberdade na criação e recolha de dados.

### **4.3 Proposta de solução**

O que se propõe é uma estrutura colaborativa de trabalho, onde, através de uma plataforma Microsoft Office SharePoint Server 2007, podemos disponibilizar uma página através de um acesso Internet, com acesso autenticado e permissões definidas a cada grupo de investigação, onde o inquiridor ou o inquirido poderá utilizar um formulário em formato digital, criado em InfoPath e onde todos os dados recolhidos serão em tempo real.

Este formulário será criado, dentro do possível, com um interface idêntico ao do formulário em papel, mas onde através de opções disponibilizadas pela ferramenta, poderemos facilitar o preenchimento e torná-lo mais dinâmico e apelativo. Através deste formulário, toda a estrutura e preenchimento de dados pode ser controlado, uniforme e fiável, pois a ferramenta introduz opções de controlo, tal como dados incorrectos ou em falha, apenas submetendo os dados se todas as condições de controlo definidas forem cumpridas. Na plataforma, o acesso permitirá que além de adicionar um formulário, o inquirido pode ainda corrigir em qualquer altura uma falha detectada nos dados de um formulário.

A criação de formulários é bastante simples, e pode ser feita sem grandes conhecimentos de programação, o que torna este processo independente de terceiros para escalar a solução, acelerar e multiplicar o processo de criação de formulários e publicação nestes e noutros projectos.

A recolha dos dados das listas de inquéritos guardados na ferramenta, será feita por um administrador da plataforma, que poderá aceder a todos os dados e realizar o processo de análise e integração a partir de várias fontes, desta vez de um modo totalmente informatizado e sem qualquer tipo de intervenção directa nos dados.

As vantagens desta nova estrutura são bastantes, pois a disponibilidade dos dados é 24/7, sem papel, sem erros, sem necessidade de deslocação de inquiridores, sem intervenção

humana directa nos dados, de rápida criação de novos formulários e com capacidade de integrar com várias plataformas e aplicações e de uma utilização muito simples e intuitiva para o utilizador.

A solução a implementar, deverá ser faseada a um grupo restrito de funcionalidades e de utilizadores, de modo a ajustar toda a capacidade do sistema. Numa fase mais avançada, serão disponibilizadas ferramentas adicionais a alguns utilizadores com conhecimentos mais avançados que serão a interface entre o desenvolvimento de novas funcionalidades e as necessidades encontradas pelos utilizadores.

O sistema permite o acesso através da ferramenta InfoPath ou por um interface web através de um browser que disponibiliza o acesso a formulários on-line para uma biblioteca de documentos no Sharepoint. Essa biblioteca permite que por meio de exportação directa para Excel ou Access, podendo ainda através de um interface de Web Service serem carregados directamente numa base de dados SQL, o que permite ter acesso aos dados prontos a serem tratados de várias formas.

Os formulários, são totalmente criados, estruturados e definidos pela administração, com processos bem definidos, que ajudam o investigador na sua utilização, e permite que todos os dados recolhidos sejam os esperados.

Deste modo, o acesso, disponibilização e tratamento da informação é totalmente centralizado, com diversas vistas, dependendo do tipo de acesso autenticado. Visto ser suportado numa Framework de gestão de conteúdos, a manutenção é suportada num sistema integrado, e não no desenvolvimento de uma aplicação á medida.

Com base nesta plataforma, o nosso principal enfoque será na funcionalidade de Forms Services, mas outros serviços do SharePoint Services irão ser utilizados em conjunto, dentro do propósito do trabalho.

## **5 Abordagem à solução**

A abordagem à solução, depois de se decidir pela utilização do MOSS, passou por analisar e pensar numa solução para as três questões derivadas que se enuncia:

1. Criação e Parametrização da plataforma SharePoint: Como instalar e parametrizar a infra-estrutura do MOSS integrada nos serviços do domínio?
2. Autenticação: Como autenticar um acesso sem integração com a Active Directory da rede do domínio?
3. Formulários: O InfoPath, explorar as suas capacidades de formulário dinâmico.
4. Estudo das soluções possíveis para a recolha e integração dos dados: Que soluções existem para integração com uma base de dados e qual a mais simples de manter e gerir?
  - Lista Sharepoint,
  - Web Service,
  - Workflows,

Neste capítulo são ainda descritos os processos de recolha de dados pelo administrador responsável pela concentração dos dados e o do utilizador responsável pelo preenchimento dos questionários online, na solução implementada.

### **5.1 Criação e parametrização da Plataforma Sharepoint**

A instalação da plataforma foi efectuada conforme os requisitos e indicações do fabricante, sendo que os requisitos mínimos estão descritos no Anexo – A.2.2 Recursos Necessários. Devido á estrutura física e lógica adoptada para a solução, foram ainda adicionadas mais algumas aplicações conforme o Anexo D.1 - Plataforma a instalar.

Ao ser instalado o Microsoft Office Sharepoint, por defeito e sem qualquer tipo de configuração por parte do administrador, cria uma configuração que utiliza os recursos envolventes a nível do domínio onde replica por defeito os utilizadores. Visto o sistema operativo estar integrado num Domínio, foi necessário quebrar essa ligação de modo a que não fossem utilizados quaisquer recursos de domínio, removendo a ligação de importação e alterando a definição da importação de perfis de utilizadores.

A parametrização correcta da plataforma irá disponibilizar a estrutura de Web collections, Websites e serviços que desejamos implementar.

A estrutura base do SharePoint assenta em Web applications, e no seu módulo de administração existem duas Web Applications comuns a qualquer instalação: o site de administração de Shared Services e o Central Administration site. A partir desta base, é possível realizar todas as configurações necessárias para desenvolver o projecto.

No site Central Administration, na página Operations podemos ter acesso às definições de:

- Topologia e serviços – Definições de serviço de mail e servidores na Farm
- Configuração Global – Gestão de features, jobs e links
- Configuração de Segurança – Antivírus, bloqueio de ficheiros, policy, etc.
- Backup e Restore – Backups e restauro da plataforma

Ainda no site Central Administration, na página Application Management podemos ter acesso a:

- SharePoint Web Application Management – Aqui podemos criar, remover, apagar ou estender uma Web application e todas as suas features.
- SharePoint Site Management – Gestão de site collections tal como criar, apagar e listar site collections.
- Application Security – Este ponto permite a administração das opções de segurança ao nível da Web application, onde é possível gerir a segurança, WebParts, self Site,

permissões de utilizadores, e configuração de authentication providers, também discutidos mais á frente no projecto.

- InfoPath Forms Services – Permite a gestão de form templates, proxy dos web services, e configuração dos Form Services.
- Workflow Management – Permite a gestão das definições de workflows.

No Site de Shared Services Administration temos as configurações do Shared Services Provider que permitem que se faça a gestão de perfis, audiências, pesquisa e Business Data Catalog.

A partir daqui, temos como site de administração Central o <http://inquerito:47272> e o Site de Shared Services com o endereço <http://inquerito:18637> com a estrutura apresentada na imagem abaixo. Estes Sites são montados com a instalação base do Sharepoint e permitem toda a administração da plataforma de um modo centralizado, simples e extensível a uma “Farm”.

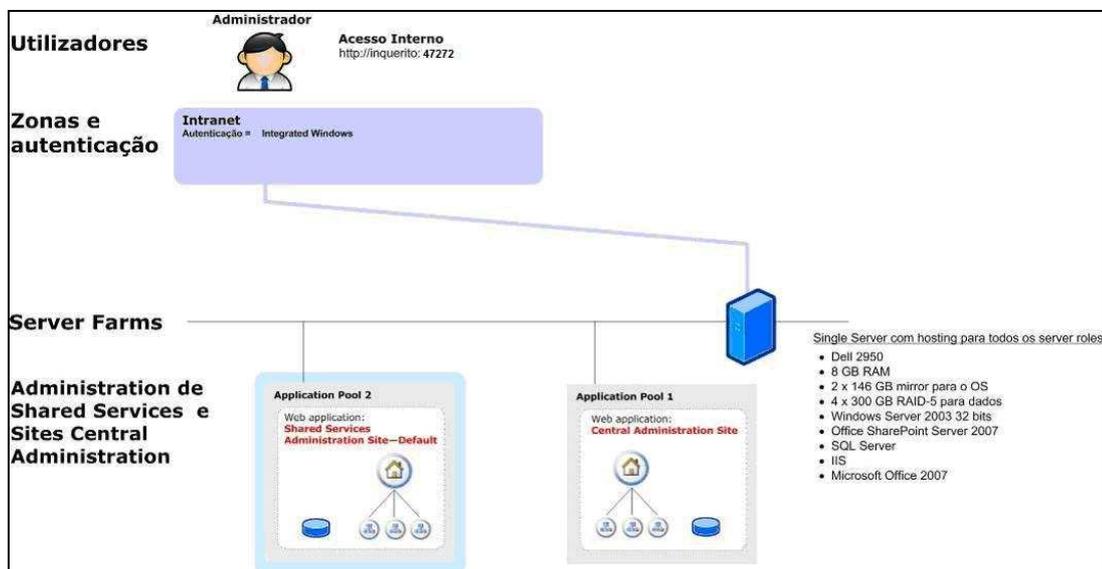


Figura 1 – Estrutura do site de Administração Central Sharepoint

Nestes módulos iremos realizar todas as tarefas de criação e configuração de site collections, Web application e segurança, necessárias á plataforma e serviços que queremos implementar.

### **5.1.1 Web Application Base**

Por defeito, o Sharepoint não permite que seja criado e disponibilizado para a Internet e Intranet um site com autenticação FBA com SQL. Logo, é necessário criar uma Web application base e estendê-la para que seja possível utilizar a autenticação definida para o projecto. Deste modo, é necessário criar uma nova Web application, e esta será a de acesso Intranet com autenticação Windows. A partir desta Web application base, iremos estender para uma nova Web application especificamente para utilizadores Internet e configurada com “forms authentication”.

Esta Web Application base terá as seguintes características

- Descrição – Extranet 2222
- Porta – 2222
- Authentication Provider – NTLM
- Permitir acesso anónimo – Não
- Load Balanced Url – http://inquerito:2222
- Database authentication – Windows authentication
- O acesso é apenas local e de administração.

A descrição relativa aos tipos de autenticação aqui descritos encontra-se no anexo A.2.7 - Windows Authentication, A.2.6 - Anonymous Authentication e A.2.8 - Forms Based Authentication

### **5.1.2 Site Collection Base**

É necessário criar uma site collection base onde temos toda a estrutura que iremos criar.

Na Application Management da Central Administration, podemos criar a Site Collection base, onde podemos definir o nome, linguagem, template de site, e utilizadores de administração. Esta é a site collection base de desenvolvimento da estrutura.

Iremos utilizar a Web application criada anteriormente e um site template no formato Publishing, e instalando a opção Publishing Portal.



**Figura 2 – Portal Inicial**

Este template disponibiliza uma hierarquia inicial para um site Internet ou um portal colaborativo. Inclui uma *home page*, um *search center* e uma página de *login*.

### 5.1.3 Internet Web Application

Este é um passo fundamental para criar o site com autenticação FBA. Visto a primeira Web application estar predefinida com a autenticação Windows, esta será a base para o site para a Zona Internet com as seguintes características:

- Web Application Seleccionada: <http://inquerito:2222>
- Descrição – inquéritos – 80
- Porta 80
- Autentication provider – NTLM
- Permitir acesso anónimo – Não

- Load Balanced Url – <http://inquerito:80>
- Database authentication – Windows authentication
- Load balanced url – <http://inquerito:80>
- Zona – Internet

Neste momento temos criada com sucesso uma site collection Extranet: <http://inquerito:2222> e um site internet <http://inquerito:80>, ambos com autenticação Windows.

#### 5.1.4 Web.config

Para que a plataforma possa reconhecer a existência de uma forma de autenticação que não é “default Windows”, é necessário alterar no Web.config<sup>2</sup> as definições para a connection string<sup>3</sup> da Base de dados SQL onde temos os utilizadores e roles criados, e toda a informação de MembershipProvider e RoleProvider. Estas alterações devem ser realizadas no Web.config da Central Administration e também em todas as Web applications existentes que necessitem desta funcionalidade.

---

<sup>2</sup> Web.config – ficheiro XML que define a configuração e definições numa aplicação Web em ASP.NET

<sup>3</sup> *Connection String* – *String* que especifica informação relativa ao acesso a uma base de dados

```

1 <connectionStrings>
2   <add name="FBA_SqlConnectionString" connectionString="server=CEIDSS\OFFICESERVERS;
3     database=FBAASpNetDb; Integrated Security=SSPI;" providerName="System.Data.SqlClient" />
4 </connectionStrings>
5 <system.web>
6   <membership defaultProvider="FBA_AspNetSqlMembershipProvider">
7     <providers>
8       <add name="FBA_AspNetSqlMembershipProvider" type="System.Web.Security.SqlMembershipProvider;
9         System.Web, Version=2.0.3600.0, Culture=neutral; PublicKeyToken=b03f5f7f11d50a3a"
10        connectionStringName="FBA_SqlConnectionString" enablePasswordRetrieval="false"
11        enablePasswordReset="true" requiresQuestionAndAnswer="false"
12        applicationName="/" requiresUniqueEmail="false" passwordFormat="Hashed"
13        maxInvalidPasswordAttempts="5" minRequiredPasswordLength="1"
14        minRequiredNonalphanumericCharacters="0" passwordAttemptWindow="10"
15        passwordStrengthRegularExpression="" />
16     </providers>
17   </membership>
18   <!-- ROLE PROVIDER -->
19   <roleManager enabled="true" defaultProvider="FBA_AspNetSqlRoleProvider">
20     <providers>
21       <add name="FBA_AspNetSqlRoleProvider" type="System.Web.Security.SqlRoleProvider;
22         System.Web, Version=2.0.3600.0, Culture=neutral; PublicKeyToken=b03f5f7f11d50a3a"
23        connectionStringName="FBA_SqlConnectionString" applicationName="/" />
24     </providers>
25   </roleManager>

```

Figura 3 - Web.config Web application base e extended

```

1 <connectionStrings>
2   <add name="FBA_SqlConnectionString" connectionString="server=CEIDSS\OFFICESERVERS; database=FBAASpNetDb;
3     Integrated Security=SSPI;" providerName="System.Data.SqlClient" />
4 </connectionStrings>
5
6 <system.web>
7   <membership defaultProvider="FBA_AspNetSqlMembershipProvider">
8     <providers>
9       <add name="FBA_AspNetSqlMembershipProvider" type="System.Web.Security.SqlMembershipProvider;
10        System.Web, Version=2.0.3600.0, Culture=neutral; PublicKeyToken=b03f5f7f11d50a3a"
11        connectionStringName="FBA_SqlConnectionString" enablePasswordRetrieval="false"
12        enablePasswordReset="true" requiresQuestionAndAnswer="false"
13        applicationName="/" requiresUniqueEmail="false" passwordFormat="Hashed"
14        maxInvalidPasswordAttempts="5" minRequiredPasswordLength="1"
15        minRequiredNonalphanumericCharacters="0" passwordAttemptWindow="10"
16        passwordStrengthRegularExpression="" />
17     </providers>
18   </membership>
19   <!-- ROLE PROVIDER -->
20   <roleManager enabled="true" defaultProvider="AspNetWindowsTokenRoleProvider">
21     <providers>
22       <add name="FBA_AspNetSqlRoleProvider" type="System.Web.Security.SqlRoleProvider;
23         System.Web, Version=2.0.3600.0, Culture=neutral; PublicKeyToken=b03f5f7f11d50a3a"
24        connectionStringName="FBA_SqlConnectionString" applicationName="/" />
25     </providers>
26   </roleManager>
27

```

Figura 4 – Web.config administração

Este ficheiro Web.config referencia a connectionString de acesso á base de dados onde estão introduzidos os utilizadores, o MembershipProvider e o roleprovider definidos para utilizar com a Base de dados.

### 5.1.5 Activação de Forms Authentication

Para que seja possível ao utilizador ter uma forma de introduzir os seus dados de acesso, é necessário que seja activada a opção de forms authentication nas opções de segurança da Web application.

Neste caso, configuramos apenas a Web application que está na zona Internet com o Membership provider e role Management definidos na criação da aplicação de autenticação de utilizadores.

Para configurar estes serviços, a Web application terá as seguintes características.

- Authentication Type – Forms
- Anonymous Access - enable
- Membership Provider – FBA\_AspNetSqlMembershipProvider
- Role Manager – FBA\_AspNetSqlRoleProvider
- Enable Client Integration – Yes

Os diferentes tipos de autenticação descritos neste capítulo estão descritos nos Anexos A.2.5 - Segurança no Sharepoint, A.2.6 - Anonymous Authentication, e A.2.8 - Forms Based Authentication

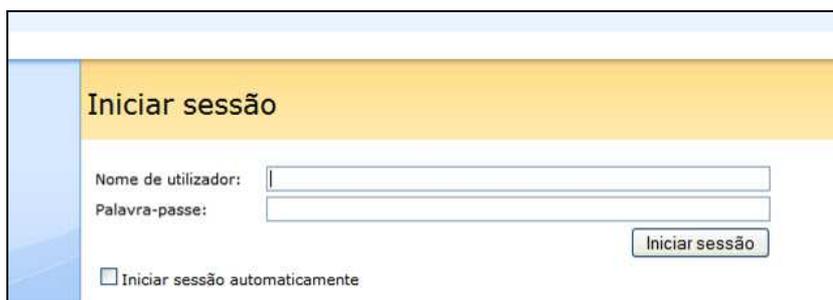
Deste modo, passamos de uma autenticação Windows, que neste caso não existe devido a não termos qualquer ligação á Active Directory, e passamos a utilizar o MembershipProvider criado com a aplicação Web e a ligação á base de dados SQL. A página de authentication provider terá as definições seguintes:

Web application	Zone	Membership provider
http://inquerito:2222	Default	Windows
	Internet	FBA_AspNetSqlMembershipProvider

**Tabela 1 – MembershipProviders Intranet e Internet**

A partir deste momento é possível aceder á Web application com uma conta de administração e adicionar os utilizadores anteriormente introduzidos com a ferramenta ASP.NET Web Site Administration Tool e definir as permissões que desejarmos dar a cada utilizador ou a grupos de utilizadores.

Se acedermos ao endereço <http://inquerito.uatla.pt> iremos verificar que já podemos ter acesso á opção Iniciar Sessão e já é possível aceder ao formulário de autenticação da plataforma, e onde podemos aceder aos sites definidos a cada grupo.

The image shows a web form for logging in. At the top, there is a yellow header bar with the text "Iniciar sessão" in black. Below this, there are two input fields: "Nome de utilizador:" and "Palavra-passe:". To the right of the password field is a button labeled "Iniciar sessão". Below the input fields, there is a checkbox labeled "Iniciar sessão automaticamente". The form is set against a light blue background with a darker blue vertical bar on the left side.

**Figura 5 - Formulário de início de sessão**

### **5.1.6 Sites dos grupos de Investigação**

A criação dos sites para cada grupo de investigação é o passo seguinte, assim, iremos criar três sites, um para cada inquérito, e com as permissões de acesso segundo a seguinte estrutura. Nesta estrutura, no Top Site definido como Home, o acesso é possível mesmo sem estar autenticado, onde poderá estar definida a visualização de informação livre disponibilizada por algumas webparts. Um utilizador não autenticado não poderá ter acesso aos sites de inquéritos. A sua navegação está restrita apenas ao Top Site Home. Apenas um utilizador com autenticação válida poderá autenticar na plataforma e ter acesso ao seu site. A definição das permissões encontra-se descrita no Anexo A.2.9 - Permissões Sharepoint.

Os seguintes sites, inq1, inq2 e inq3 serão criados com os endereços segundo a estrutura da tabela seguinte.

Zona	Endereço	Site	Grupo	Autenticação	Permissões
Internet	<a href="http://inquerito.uatla.pt">http://inquerito.uatla.pt</a>	Home Top Site		Anonymous	Read
	<a href="http://inquerito.uatla.pt/inq1">http://inquerito.uatla.pt/inq1</a>	Inquérito1	Grupo 1	FBA	Contribute
	<a href="http://inquerito.uatla.pt/inq2">http://inquerito.uatla.pt/inq2</a>	Inquérito2	Grupo 2	FBA	Contribute
	<a href="http://inquerito.uatla.pt/inq3">http://inquerito.uatla.pt/inq3</a>	Inquérito3	Grupo 3	FBA	Contribute
Intranet	<a href="http://inquerito:2222">http://inquerito:2222</a>			Windows	Administração

Tabela 2 - Estrutura Sites de Inquérito

Dentro de cada site será adicionado o template de biblioteca de documentos do tipo formulário, definindo assim o tipo de conteúdo da biblioteca. Esta biblioteca apenas disponibilizará o formulário que for definido pela administração.

O utilizador poderá através do seu browser, criar um novo formulário, ter acesso a uma lista definida para visualizar apenas os inquéritos realizados pelo utilizador autenticado ou abrindo um formulário dessa lista, ter acesso a todos os dados que introduziu, podendo inclusive alterar e corrigir os dados se necessário.

Teremos também disponível um site de pesquisa interna na plataforma, onde um utilizador pode facilmente encontrar um formulário criado por si, ou qualquer tipo de informação que se pretenda disponibilizar para ser pesquisada.

Será ainda criado um site de suporte e ajuda ao acesso, funcionamento, estrutura e preenchimento dos formulários.

Deste modo, podemos assumir esta estrutura simplificada que se pretende disponibilizar para o propósito de recolha de dados em formulários conforme o gráfico abaixo.

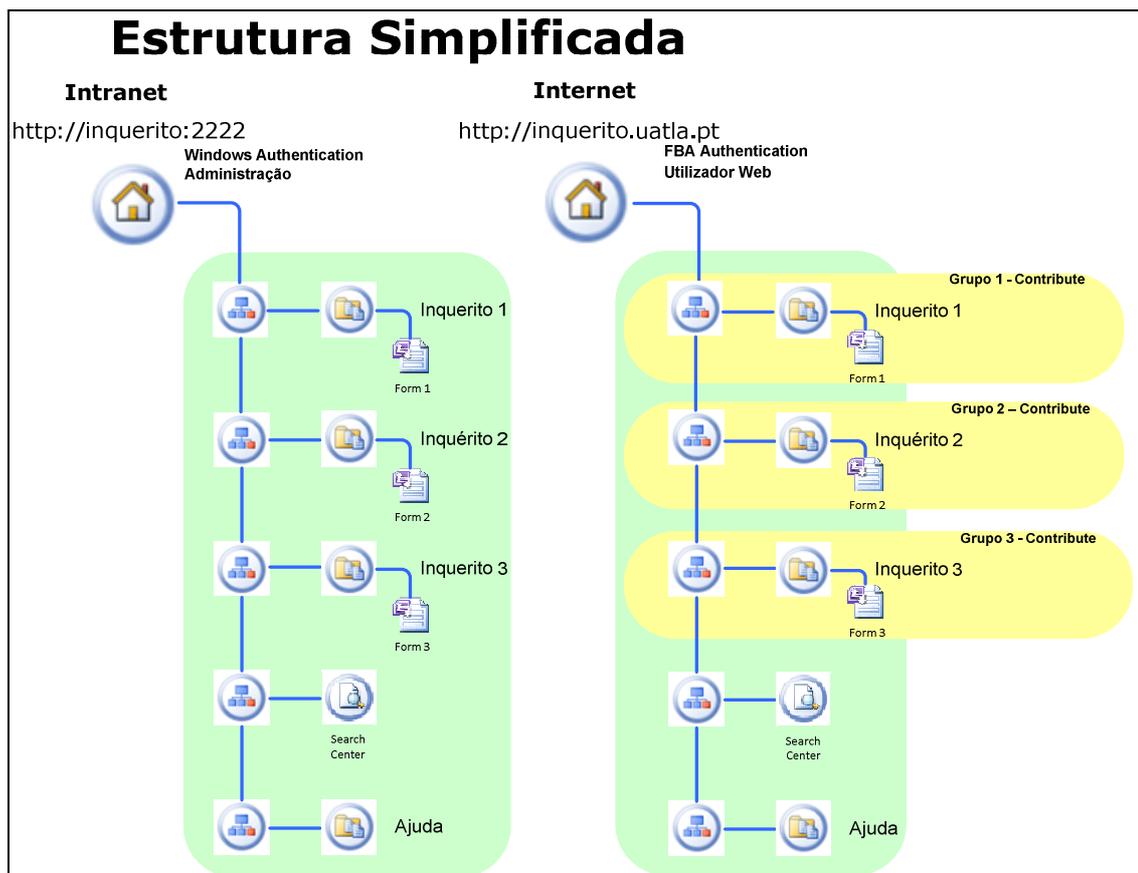


Figura 6 – Estrutura da Plataforma instalada

## 5.2 Autenticação

Por defeito, a plataforma permite que seja utilizada a Active Directory, onde a criação de utilizadores é bastante simples na criação e gestão. Infelizmente, devido a restrições definidas pela equipa de Administração de Sistemas da Organização não foi possível utilizar a AD já existente ou que fosse criada uma AD específica para o projecto.

Foi assim necessário encontrar uma solução que se adaptasse às características do projecto e permitisse uma gestão simples e eficiente que permitisse o mesmo tipo de estrutura e funcionalidade que uma AD. A solução encontrada foi a utilização da autenticação utilizando a plataforma ASP.NET 2.0 e uma base de dados SQL. Todas a informação relativa aos métodos de autenticação referidos nos pontos seguintes estão descritos nos Anexos A.2.5 – Segurança no Sharepoint, A.2.6 – Anonymous Authentication, A.2.7 - Windows Authentication e A.2.8 - Forms Based Authentication

Visto existir um acesso Internet, o acesso á plataforma nos Top Sites estará disponível para todos, o que é possível pois o acesso é livre na página principal com anonymous authentication, mas para aceder a qualquer site de inquérito ou funcionalidade adicional é necessário um utilizador autenticado. A anonymous authentication, servirá apenas no propósito de permitir o acesso á página principal, onde a partir dai o utilizador se pode autenticar para aceder ao seu site com o formulário, ou pode apenas navegar nas páginas e Top Sites assim definidos.



**Figura 7 – Página de entrada sem autenticação**

Nestas condições, além de permitir a navegação livre, é necessário um interface onde o utilizador se possa autenticar, o que é disponibilizado por formulário de acesso integrado com Forms Based Authentication (FBA) com base numa BD SQL dará acesso ao site que irá utilizar para preencher os seus formulários, Finalmente, Windows authentication será configurada na estrutura de administração, onde será definido full control nas aplicações e administração. Apenas é acessível dentro do servidor, com utilizador de administração. Estas opções derivam dos requisitos definidos para os utilizadores e administradores, visto que os utilizadores, devem poder ter acesso ao sistema, tendo ou não instalado o Microsoft InfoPath, sendo que essa opção apenas é possível pelo interface Web, onde este tipo de autenticação se adequa, pois permite a gestão de Grupos e utilizadores na plataforma.

### **5.2.1 Autenticação FBA e SQL**

A opção mais utilizada a nível de autenticação no SharePoint, é sem dúvida a autenticação Windows numa Active Directory, pois além de toda a estrutura de utilizadores, Grupos, OUs já estar definida, a integração é total, fácil de gerir e simples de implementar, mas visto não ser possível, optamos pela autenticação FBA (Forms Based Authentication) por ser a que melhor se adequa às necessidades do projecto.

A Framework ASP.NET 2.0 inclui a class membership que permite autenticar, adicionar, criar, remover e modificar contas de utilizadores com todo o suporte e customização no Web.config da aplicação Web, de modo a que, através de uma DB SQL ou através da Active Directory seja possível criar um interface fiável.

Relativamente ao SQL, temos o SqlMembershipProvider que permite utilizar uma BD para guardar toda a informação de autenticação. Este membershipProvider provider cria um layer que permite que, por exemplo o login control e os dados na data store possam ser utilizados. A informação relativa a este tipo de autenticação encontra-se no Anexo A.2.8 - Forms Based Authentication

É ainda possível através desta Framework criar e administrar Roles com a classe RoleProvider, o que permite que sejam definidos grupos de utilizadores, tal como numa AD, e criar regras de permissões, onde podemos dar ou limitar o acesso apenas com base no role definido do utilizador. É assim possível restringir toda a funcionalidade de permissões na aplicação.

### **5.2.2 Criação da Base de dados FBAASPNetDB**

Por defeito, o Sharepoint não tem nenhuma opção de autenticação FBA numa base de dados SQL. Para tal foi necessário criar todo um conjunto de recursos para disponibilizar esta funcionalidade.

Em primeiro lugar foi utilizada uma ferramenta Microsoft para criar a base de dados e toda a sua estrutura, o aspnet\_regsql.exe, disponível em %windir%\Microsoft.Net\Framework\v2.0.50727. Esta aplicação permite criar rapidamente toda a base de dados necessária, conforme se pode verificar na imagem seguinte.

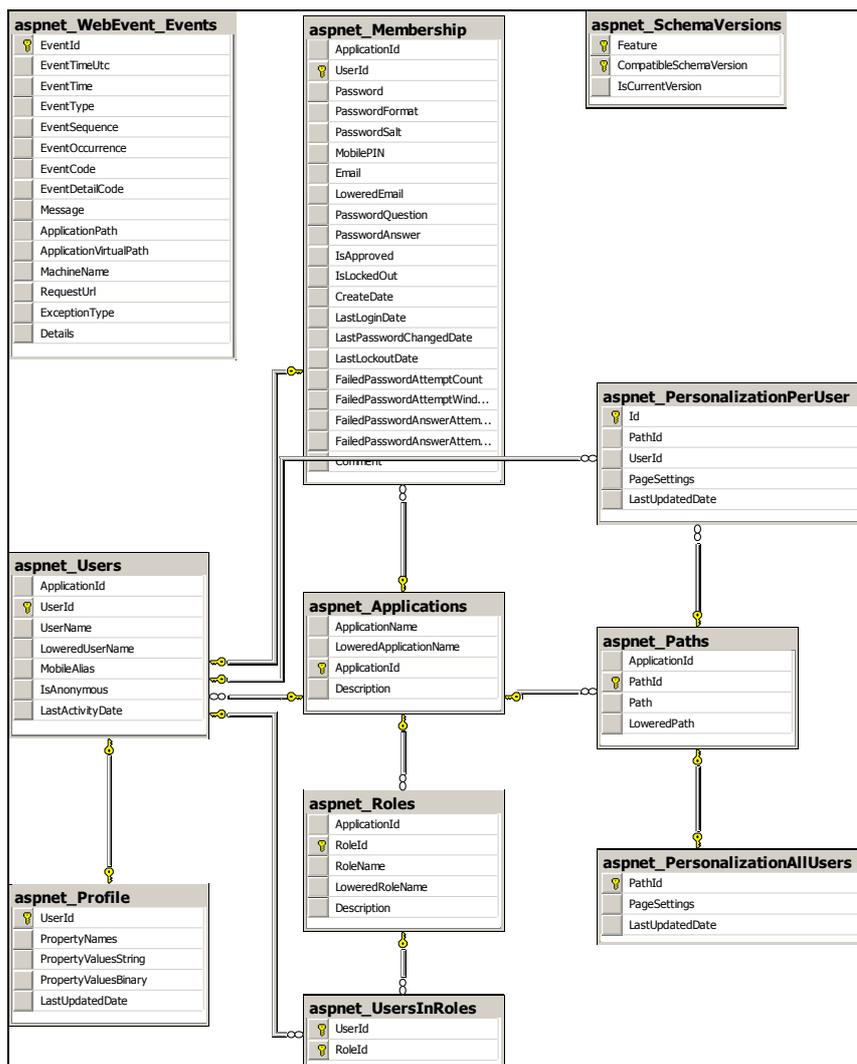


Figura 8 – Diagrama da Base de Dados de autenticação

### 5.2.3 Aplicação Web de gestão de users

Na continuação do ponto anterior, para permitir que sejam criado e geridos os utilizadores de modo a utilizarmos autenticação FBA, é necessário aceder a um interface de gestão, que já existe na Framework.

Para tal, utilizando o Visual Studio 2005, foi criada uma pequena aplicação Web para aceder ao interface.

O ponto mais importante deste processo, é a configuração do Web.config onde é definida a connectionString, membership defaultProvider, roleManager, defaultProvider e o authentication mode segundo o código abaixo<sup>4</sup>:

```
<appSettings/>
<connectionStrings>
  <add name="FBA_SqlConnectionString"
    connectionString="server=localhost\OFFICESERVERS;
    database=FBAASPNetDB; Integrated Security=SSPI;"
    providerName="System.Data.SqlClient"/>
</connectionStrings>
....
<system.web>
  <membership defaultProvider="FBA_AspNetSqlMembershipProvider">
    <providers>
      <add connectionStringName="FBA_SqlConnectionString"
        enablePasswordRetrieval="false" enablePasswordReset="true" requiresQuestionAndAnswer="false"
        applicationName="/" requiresUniqueEmail="false" passwordFormat="Hashed"
        maxInvalidPasswordAttempts="5" minRequiredPasswordLength="1"
        minRequiredNonalphanumericCharacters="0" passwordAttemptWindow="10"
        passwordStrengthRegularExpression="" name="FBA_AspNetSqlMembershipProvider"
        type="System.Web.Security.SqlMembershipProvider, System.Web;Version=2.0.3600.0,
        Culture=neutral; PublicKeyToken=b03f5f7f11d50a3a" />
    </providers>
  </membership>

  <roleManager enabled="true" defaultProvider="FBA_AspNetSqlRoleProvider">
    <providers>
      <add connectionStringName="FBA_SqlConnectionString" applicationName="/"
        name="FBA_AspNetSqlRoleProvider" type="System.Web.Security.SqlRoleProvider, System.Web;
        Version=2.0.3600.0, Culture=neutral; PublicKeyToken=b03f5f7f11d50a3a" />
    </providers>
  </roleManager>
</system.web>
....
<!--
```

Figura 9 – Web.config da Web application de autenticação

A partir deste ponto, é possível aceder ao módulo de configuração ASP.NET pela ASP.NET configuration, onde iremos criar os acessos e definir o Membership Provider e o Role Provider. Através da opção Create User, podemos criar um utilizador configurar e confirmar a *password* e uma conta de mail. O processo é muito semelhante na criação de Roles e atribuição dos mesmos a utilizadores. A nível da criação e gestão de utilizadores, estes são os pontos necessários á implementação.

Mais tarde na definição de permissões, estes Roles e utilizadores serão configurados na plataforma para definir o tipo de possibilidade de tarefas que podem realizar.

---

<sup>4</sup> Código-Fonte disponível em anexo



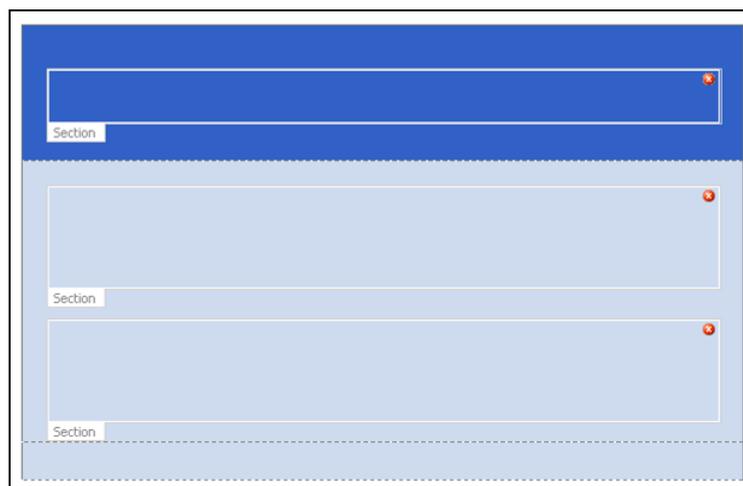
Figura 10 – Configuração Provider e criação de utilizador

### 5.3 O InfoPath na criação de formulários

Todos os formulários foram criados com a ferramenta Microsoft InfoPath 2007 devido às diferentes possibilidades oferecidas pela mesma, conforme o que referimos no Anexo C.

Foi criado um formulário de base Web definido em layout único, onde se tem a vista completa do corpo completo do formulário, tal como o formulário em papel, mas onde, sempre que possível, se introduziram algumas funcionalidades de modo a facilitar o preenchimento e a tornar o próprio formulário mais dinâmico.

O formulário é composto por um *layout* com duas frames, uma primária de título e uma secundária onde são desenhados todos os componentes de dados a preencher. Em cada frame foram criadas sections onde ficarão delimitados no formulário os componentes que serão criados. Na segunda frame, foram criadas várias sections conforme a imagem seguinte:

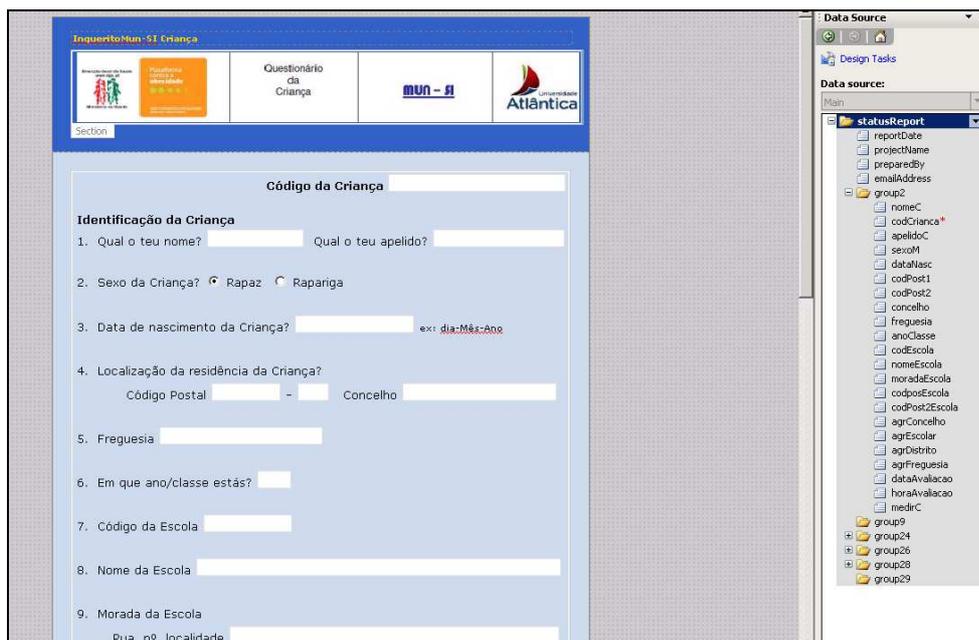


**Figura 11 - Layout do formulário**

Nestas sections são introduzidos controlos como texto, textboxes, dropdown List box, checkbox ou option buttons, etc. Cada um destes controlos tem um Datasource associado e é nestes Datasources que os dados são armazenados para serem enviados para a plataforma ou web service. Como podemos ver na imagem abaixo, cada controlo como nome, código criança, ou qualquer outro controlo que possa conter qualquer tipo de dados, tem um datasource correspondente. Por exemplo, o controlo código da criança tem as seguintes propriedades:

Controlo	textbox field
Field name	codCrianca
Data type	Text (String)
Datasource	codCrianca
Validação	Não pode ser null

**Tabela 3 - Características de field em InfoPath**



**Figura 4 - DataSource do formulário**

Nestes controlos são definidos os tipos de dados, como texto, data, inteiro, etc, e permite ainda que se definam regras e condições de validação de dados introduzidos. Deste modo podemos garantir que os dados que se irão introduzir serão os esperados e no caso de isso não acontecer o formulário assinalará o controlo com erro para que o utilizador o corrija e nunca será permitido que submeta o formulário enquanto não forem corrigidos todos os erros.

### 5.3.1 Business Logic do Formulário

Os eventos no formulário, permitem ao utilizador escolher uma determinada opção de uma checkbox e uma section poderá ficar de imediato indisponível ou invisível, e todos os valores dos controlos que possam ter sido introduzidos podem ser alterados.

A isto chama-se Business Logic do InfoPath, onde se controla opções do formulário através de regras, validação de dados, formulas ou código, criando um formulário dinâmico em que é possível um elevado grau de controlo de erros.

De modo a tornar a utilização e preenchimento do formulário mais simples e intuitivo, foi definida uma Business Logic em que alguns blocos de questões do formulário não estariam

visíveis com base no valor de alguns controlos. Deste modo, o utilizador tem um acesso mais rápido às questões, sem “saltos” durante o preenchimento, e onde o que ele vê é apenas o que necessita de ser preenchido.

Como exemplo e utilizando a Business Logic implementada, com base nos valores dos controlos quando seleccionados, podemos criar uma formatação condicional em que se uma checkbox for seleccionada, uma section do formulário fica sem poder ser visualizada ou passa a ser visualizada. Seja numa opção em que uma section se torna visível ou invisível, todo o formulário se ajusta automaticamente á nova secção ou escondendo as que desejarmos tornando as seguintes acessíveis.

Na imagem seguinte podemos verificar que a questão nº 15 só está disponível se a opção da questão anterior for negativa. Deste modo, o utilizador do formulário, ao escolher uma opção que nega toda uma série de questões, terá acesso apenas ao caminho que escolheu, tornando o formulário mais simples e uma experiência mais dinâmica.

Vista sem validação condicional	Vista com validação condicional
<p>13. Hora da Avaliação? <input type="text"/> ex: hh:mm</p> <p>14. Agora, gostaria de avaliar o teu peso, medir a tua altura, medir a tua cintura e a anca. Vou explicar-te como vou fazê-lo. Posso passar a medir-te?</p> <p><input checked="" type="radio"/> Sim; a criança concorda em ser avaliada ( após a validação continue com a pergunta 16)</p> <p><input type="radio"/> Não; a criança não concorda em ser avaliada (complete a pergunta 15, coloque o seu código (22) nome (23)).</p> <p><b>Itens para Avaliação</b></p> <p>16. Peso <input type="text"/></p> <p>17. Altura <input type="text"/> cm - <input type="text"/> cm</p> <p>18. Peso (10% de fiabilidade) <input type="text"/> kg</p> <p>19. Altura (10% fiabilidade) <input type="text"/> cm <input type="text"/> cm</p> <p>20. Descreva as roupas que a criança tem vestidas quando está a ser avaliada. por favor, lembre-se de retirar o calçado, meias ou collants assim como qualquer objecto (telemóvel, carteira, cinto, ganchos/outros adereços de cabelo, etc.)</p> <p><input type="radio"/> Apenas roupa interior</p> <p><input type="radio"/> Roupa de ginástica (por ex. apenas calções e t-shirt)</p> <p><input type="radio"/> Roupa leve (por ex. t-shirt ou saia de algodão)</p>	<p>13. Hora da Avaliação? <input type="text"/> ex: hh:mm</p> <p>14. Agora, gostaria de avaliar o teu peso, medir a tua altura, medir a tua cintura e a anca. Vou explicar-te como vou fazê-lo. Posso passar a medir-te?</p> <p><input type="radio"/> Sim; a criança concorda em ser avaliada ( após a validação continue com a pergunta 16)</p> <p><input checked="" type="radio"/> Não; a criança não concorda em ser avaliada (complete a pergunta 15, coloque o seu código (22) nome (23)).</p> <p>15. Podes dizer-me porque não queres ser avaliado?</p> <p><input type="checkbox"/> A criança não se sente bem ou sente dor.</p> <p><input type="checkbox"/> A criança está ansiosa/nervosa.</p> <p><input type="checkbox"/> A criança tem uma deficiência física.</p> <p><input type="checkbox"/> Outra razão</p> <p>por favor especifique: <input type="text"/></p> <p>22. Código do Examinador <input type="text"/></p> <p>23. Nome do Examinador <input type="text"/></p> <p>Data 07-11-2009</p> <p>Observações: <input type="text"/></p>

Figura 12 Vistas de validação

No caso de terem sido preenchidos controlos numa section e que mais tarde foi alterada para uma opção em que é tornada inacessível essa mesma section, foi definida na Business Logic que todos os dados que foram introduzidos nessas sections sejam alterados para valores

nulos, deste modo garantindo que exista controlo nos dados preenchidos que são enviados para a plataforma.

Na figura seguinte podemos visualizar uma parte da Business Logic implementada na tabela anterior onde regras, sob determinados valores verificados em condições dos controles, efectuam acções onde, neste caso, fazem o controlo de dados de sections ocultas.

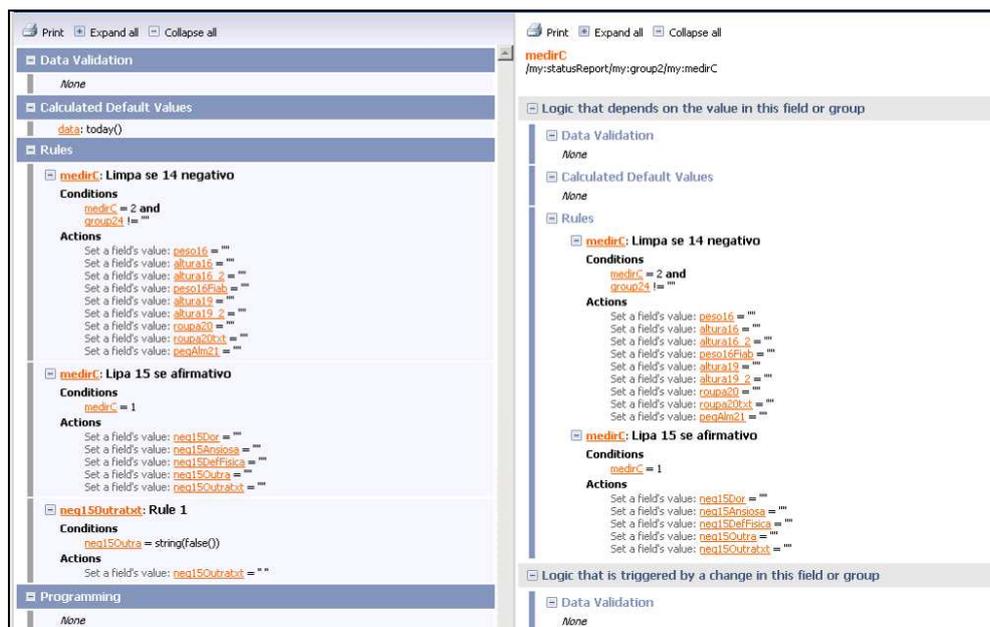


Figura 13 – Business Logic do Formulário

## 5.4 Soluções para a integração de dados

As possibilidades do SharePoint são várias, podendo ser um formulário disponibilizado para a Web ou para uma Intranet, para um browser ou para uma aplicação cliente InfoPath ou mesmo para o email.

Sendo o objectivo principal a atingir a total disponibilidade e que o utilizador possa sempre aceder e utilizar o serviço, independentemente das condições, a recolha dos dados dos formulários foi pensada para ser realizada em tempo real através do browser com acesso Internet. A utilização de uma biblioteca de formulários e listas de Sharepoint segundo uma análise das suas características conforme os anexos A.2.12 - Tipos de conteúdos e funcionalidades, A.2.13 - Site Content Type, A.2.14 - Bibliotecas e Listas de Sharepoint

serviu para, a nível do SharePoint, para armazenar os formulários preenchidos em forma de lista.

Uma vez recolhidos os dados surge a questão: Que soluções existem para integração desses dados com uma base de dados e qual a mais simples de manter e gerir?

- Lista Sharepoint de integração nativa com o Access ou Excel
- Web Service, para servir a ligação a um SQL Server
- Workflows, para automatizar a inserção de Dados numa Base de Dados externa.

### 5.4.1 O armazenamento de dados em Listas Sharepoint

Um dos requisitos deste trabalho foi que o acesso aos formulários não dependeria do sistema operativo nem das aplicações do utilizador da plataforma. Um utilizador apenas necessitaria de um browser e acesso á internet para que pudesse enviar os seus dados.

Deste modo, foram criadas bibliotecas de documentos do tipo formulário que se descrevem no anexo A.2.14 - Bibliotecas e Listas de Sharepoint, e onde é possível guardar e organizar todos os formulários criados nessa biblioteca na plataforma, conforme a imagem seguinte:

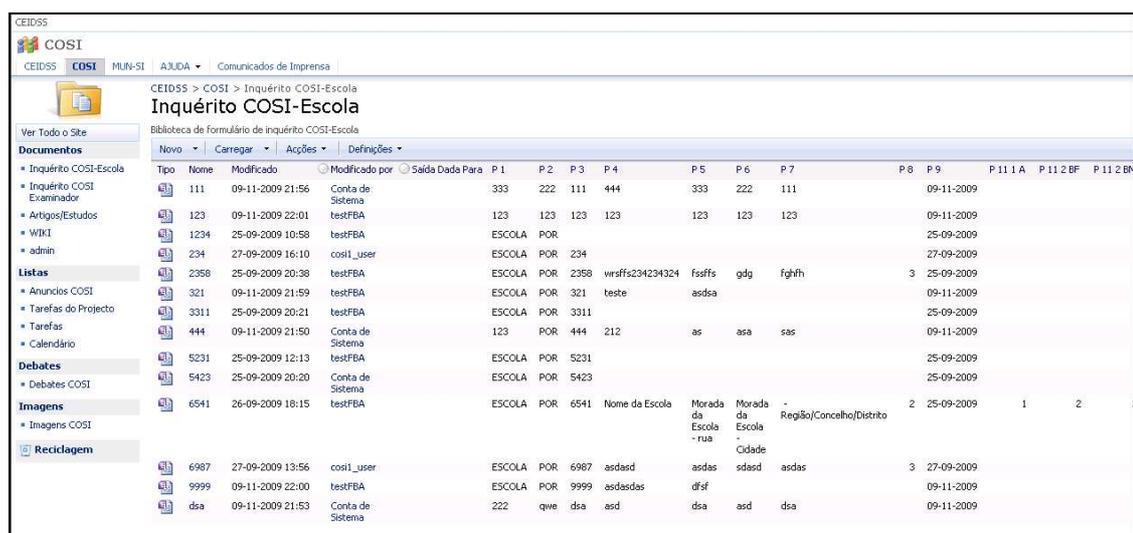


Figura 14 - Vista Biblioteca de Documentos

Ao aceder á biblioteca de formulários, o utilizador tem acesso a uma lista completa com todos os dados dos formulários enviados, onde pode criar um novo formulário através do seu browser, e depois de preencher os dados conforme as necessidades, ao clicar na opção “Submeter”, será enviado o ficheiro .XML para a biblioteca e os dados validados anteriormente no formulário serão carregados directamente para lista de Sharepoint.

Para isso, as definições de compatibilidade do formulário devem ser predefinidas na criação para utilização Web com InfoPath Forms Services ou com cliente InfoPath, pelo que foi definido o template de formulário deverá ser compatível com browser.

Apesar de utilizando a opção Web browser as opções serem um pouco mais limitadas a nível de opções de controlos nos formulários, decidimos que era adequado às necessidades.

Conforme a estrutura definida anteriormente, foi criada uma biblioteca de formulários para cada site de projecto, e será esta a biblioteca que será definida no formulário a publicar.

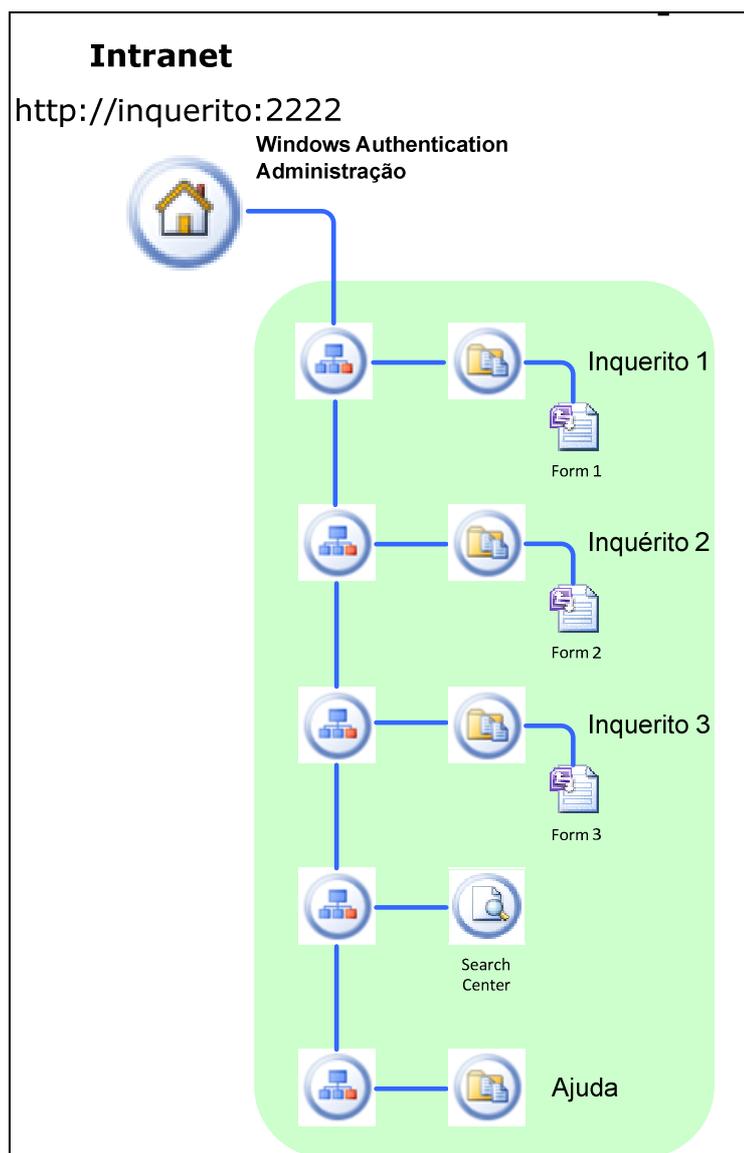


Figura 15 - Bibliotecas de Formulário

Ao definir uma solução de submeter os formulários para uma biblioteca de formulários, necessitamos de configurar no formulário as definições correctas para que o formulário tenha todas as funcionalidades pretendidas.

No formulário podem ser definidas opções base, tanto a nível da interface criada como na definição do servidor ou compatibilidade cliente/browser e foram definidas as seguintes:

Categoria	Opção	Definição
Browser	Toolbars	Toolbar no topo e no fim do formulário com as seguintes opções: Submeter, Guardar, Fechar
Linguagem	Português	
Assinaturas Digitais	Sem assinaturas digitais	
Segurança	Automático	O formulário tem acesso a conteúdo no mesmo Domínio
Compatibilidade	Browser ou cliente InfoPath	url: <a href="http://inquerito:2222/munsi/vti_bin/FormsServices.asmx">http://inquerito:2222/munsi/vti_bin/FormsServices.asmx</a>
Property Promotions	Fields	Todos os controlos definidos no formulário serão passados para uma lista na biblioteca
Controlo de versões	Upgrade automático	

Tabela 5 - Opções do formulário

Para submeter os dados do formulário, é necessário que sejam criadas opções de Submit, onde podemos definir diferentes destinos e no âmbito deste capítulo, definimos que os dados teriam como destino uma biblioteca de formulários. Deste modo, o formulário é definido com uma Data Connection de nome Main Submit onde se define como e para onde serão enviados os dados do formulário, com as seguintes características:

Nome	Main submit
Tipo	Submit Data (Apenas submete dados)
Lista ou biblioteca Sharepoint	<a href="http://inquerito:2222/munsi/MUNSI Crianca/">http://inquerito:2222/munsi/MUNSI Crianca/</a>
Nome do ficheiro .XML a criar	Com base no valor do datasource codCrianca (que identifica univocamente cada formulário)
Permitir overwrite	Se for ficheiro pré-existente

Tabela 6 - Características de Main submit

Com a configuração definida conforme a tabela anterior, definimos que uma data connection do tipo Submit Data irá enviar os dados .XML para a biblioteca de formulários no endereço <http://inquerito:2222/munsi/MUNSIcrianca/>, onde será criado um novo documento .XML cujo nome terá por base o valor do datasource definido no controlo codCrianca, e se esse valor já existir na biblioteca, substituirá o existente na biblioteca pelo novo formulário, garantindo assim a actualização do mesmo.

Ao submeter um novo formulário, tanto o ficheiro XML como os valores de cada um dos campos preenchidos do formulário serão passados através da property promotion configurada no formulário, para itens da lista disponível na biblioteca, ficando assim imediatamente disponíveis para análise.

Com esta configuração é possível ter a funcionalidade de criar um novo formulário publicado na biblioteca, que poderá ser utilizado por todos os utilizadores.

Esta foi a solução adoptada, pois através desta biblioteca o utilizador poderá criar um novo formulário, ou visualizar um já submetido, criar ou alterar vistas, sempre num interface web bastante simples e acessível, e principalmente é a opção mais simples a nível de manutenção, pois é totalmente desenhada na ferramenta InfoPath e suportada nativamente pelo Sharepoint, sem que seja necessário efectuar qualquer tipo de desenvolvimento adicional, mesmo em caso de alterações. Assim a manutenção passa a ser um processo administrativo e totalmente independente de terceiros, o que facilita e agiliza a manutenção e criação de novos formulários na plataforma.

#### **5.4.2 A integração nativa com Listas de Sharepoint**

As listas no Sharepoint podem ser importadas por uma BD Access com a mesma facilidade com que se importa um ficheiro Excel ou .XML pelo que esta solução de integração sendo nativa apenas depende do processo manual que o responsável pela acumulação dos dados registados nos formulários venha a ter. Este processo está documentado na secção 5.5.2 – Análise dos dados recolhidos – Interface Administração

### 5.4.3 Utilizar Web Services na integração de dados

Utilizar um Web Service para efectuar a inserção dos dados directamente numa base de dados SQL foi uma solução testada mas com o inconveniente de que, no futuro, seria muito mais difícil de manter, pois implica em caso de qualquer tipo de alteração a nível dos dados no formulário, que essa alteração se reflecta tanto no Web Service como na base de dados SQL.

A integração directa de um formulário Web de InfoPath e uma base de dados SQL não é possível devido a questões de segurança que se encontram descritas no Anexo C.2 – Integração InfoPath e SQL Server.

Deste modo, foi definido que para esta solução seria criado um web service no mesmo servidor de instalação da plataforma Sharepoint, num site/web application no mesmo serviço IIS (*Internet Information Services*) mas totalmente independente da plataforma, que permite que sejam feitas queries de pesquisa, inserção, updates e remoção de dados de uma base de dados instalada num servidor SQL, conforme a arquitectura definida no Anexo D - Plataforma a instalar na figura. A base de dados será criada num servidor com Microsoft SQL Server 2005 com a instância CEIDSS. A base de dados terá o nome de “Dados” e será composta apenas por uma tabela de nome “Inquerito” onde serão inseridos todos os dados e terá as seguintes características:

Nome da Coluna	Tipo de dados	Permitir nulos
idForm	INT	Não
codPais	VARCHAR(50)	Sim
codEscola	INT	Sim
nomeEscola	VARCHAR(50)	Sim
ruaEscola	VARCHAR(50)	Sim
cidadeEscola	VARCHAR(50)	Sim
regiaoEscola	VARCHAR(50)	Sim

**Tabela 7 - Características da tabela da base de dados Inquerito**

A base de dados será bastante simples, pois apenas desejamos demonstrar a funcionalidade desta solução. O script SQL de criação da Base de dados e da tabela está disponível no Anexo E.1 - Script SQL.

As vantagens da solução utilizando um web service: A funcionalidade que pretendemos é totalmente independente a nível de plataformas, linguagens de programação ou sistemas operativos. O acesso é feito por http, expondo todas as funcionalidades de programação na Web, e totalmente transparente para o utilizador.

O formulário continuará a ser enviado para uma lista de Sharepoint, tal como referido, de modo a permitir que seja facilmente pesquisado e analisado pelo utilizador ou administrador. Este passo não é absolutamente necessário, pois o web service pode ser utilizado tanto para o envio de informação para a base de dados como para a leitura da base de dados e eles serem disponibilizados do formulário, tendo sido inclusive desenvolvidos os métodos de leitura dos dados no web service, mas que por opção de desenvolvimento, não serão utilizados, sendo o carregamento dos dados de um formulário já existente todo realizado a partir da Biblioteca de formulários.

Toda a estrutura de layout será mantida a nível de controlos, Field Names, Data Types, Datasources e Validação, assim como toda a Business Logic do Formulário conforme a criação do formulário já referida nos Capítulos 5.3 – O InfoPath na criação de formulários e 5.3.1- Business Logic do Formulário.

Para realizar as operações de inserção, actualização e remoção, foram implementados diferentes métodos no Web Service, que se descrevem a seguir.

#### **5.4.3.1 Web Service – Operação de Insert na base de dados**

Apenas para efeitos de análise e devido á extensão de dados nos formulários, iremos aqui referir como dados de formulários a serem tratados pelo web service e pela base de dados apenas os sete primeiros campos do formulário de inquérito COSI.

Foi criado um web service em C# com o Microsoft Visual Studio 2008 para realizar as tarefas de inserção, remoção e actualização dos dados na tabela da base de dados do

servidor. O web Service foi desenvolvido em .NET 2.0 e instalado no mesmo IIS que estava a ser utilizado pela plataforma com as seguintes características principais:

Descrição	Webservice
Porta TCP	8090
Directório	C:\Inetpub\WebService
Permissão execução	Scripts e executáveis
Application pool	DefaultAppPool
Default content Page	Service1.aspx
Versão ASP.NET	2.0
Autenticação	Anonymous Access

**Tabela 8 - Características IIS do Web Service**

O código-fonte em linguagem C# encontra-se no E.1.3 - Código-fonte C# do Web Service.

A grande diferença de se utilizar um web service está nas Data Connections, que neste caso serão activadas por Rules nas Submit Options. Em vez de enviarmos directamente para a biblioteca de formulários, iremos criar uma Rule com o nome SEND para cada operação a realizar e cada Rule terá duas acções. Cada uma destas acções será de submit do formulário, onde uma acção enviará o formulário através de uma data connection de nome Sharepoint Library Submit para uma biblioteca, tal como no capítulo 5.4.1 – O armazenamento de dados em Listas Sharepoint e uma outra enviará o formulário para um web service através de uma data connection de nome Web Service Submit.

Nome	Sharepoint Library Submit
Tipo	Submit Data (Apenas submete dados)
Lista ou biblioteca Sharepoint	<a href="http://inquerito:2222/munsi/MUNSI Crianca/">http://inquerito:2222/munsi/MUNSI Crianca/</a>
Nome do ficheiro .XML a criar	Com base no valor do datasource codCrianca (que identifica univocamente cada formulário)

**Tabela 9 - Características de Sharepoint Library Submit**

Nome	Web Service Submit
Tipo	Submit Data (Apenas submete dados)
Lista ou biblioteca Sharepoint	<a href="http://inquerito:8090/service1.aspx">http://inquerito:8090/service1.aspx</a>
ID	Com base no valor do datasource codCrianca (que identifica univocamente cada formulário)
Operação	AddNewInq

**Tabela 10 - Características de WebService Submit**

Deste modo, ao enviar o pedido de submit do formulário, activamos a Rule Send, que está associada a duas acções, uma de envio para uma biblioteca de formulários, e uma outra para o Web service.

Durante a configuração da acção, é necessário definir qual a operação que iremos utilizar no web service, neste caso é a AddNewInq, cujo código se encontra no Anexo E.1.3 – Código-fonte C# do Web Service, e esta operação espera receber em cada variável do método, os valores respectivos do datasource que enviamos conforme o quadro abaixo:

Parâmetro (web service)	Tipo	Elemento (form)
Tns:idForm	Int	/my:statusReport/my:group37/my:p1
Tns:codPais	String	/my:statusReport/my:group37/my:p2
Tns:codEscola	Int	/my:statusReport/my:group37/my:p3
Tns:nomeEscola	String	/my:statusReport/my:group37/my:p4
Tns:ruaEscola	String	/my:statusReport/my:group37/my:p5
Tns:cidadeEscola	String	/my:statusReport/my:group37/my:p6
Tns:regiaoEscola	String	/my:statusReport/my:group37/my:p7

**Tabela 11 – Passagem de parâmetros para o web service: método addNewInq**

Os dados do formulário, estão deste modo associados a cada parâmetro do web service, e a partir daqui toda a funcionalidade é realizada por parte do web service que recebe os parâmetros e os associa a uma string SQL e os insere directamente na Base de dados.

Assim, a biblioteca recebeu os dados enviados na primeira acção da Rule, e o web service recebeu os parâmetros e inseriu-os na base de dados SQL, e isto de um modo totalmente transparente para o utilizador.

#### 5.4.3.2 Web Service - Operação de Update na base de dados

A operação de Update dos dados na Base de dados é muito semelhante á operação de Insert, e a sua grande diferença está no método de update do web service e na possibilidade de efectuar igualmente um update na biblioteca de formulários.

O formulário é praticamente idêntico, exceptuando a criação de um botão no fim do formulário a que ficará associada a Rule de Update. Este botão necessita ser criado pois não é possível associar as duas acções a uma operação de update num formulário web.

Deste modo, para realizar um update, teremos também de criar uma Rule de nome Update. Esta Rule terá duas acções. A acção de submit para a biblioteca de formulários terá o nome de Sharepoint Library Update e será idêntica á do capítulo 5.4.1 – O armazenamento de dados em Listas Sharepoint uma outra com o nome de Web Service Update enviará o formulário para um web service com a operação de UpdateInq conforme as tabelas seguintes.

Nome	Sharepoint Library Update
Tipo	Submit Data (Apenas submete dados)
Lista ou biblioteca Sharepoint	http://inquerito:2222/munsi/MUNSICrianca/
Nome do ficheiro .XML a criar	Com base no valor do datasource codCrianca (que identifica univocamente cada formulário)
Permitir overwrite	Se for ficheiro pré-existente

Tabela 12 - Características de Sharepoint Library Update

Nome	Web Service Update
Tipo	Submit Data (Apenas submete dados)
Lista ou biblioteca Sharepoint	<a href="http://inquerito:8090/service1.aspx">http://inquerito:8090/service1.aspx</a>
ID	Com base no valor do datasource codCrianca (que identifica univocamente cada formulário)
Operação	UpdateInq

Tabela 13 - Características do Web Service Update

Na data connection Sharepoint Library Update, a diferença para o submit está na opção de permitir o overwrite se o ficheiro já existir, deste modo garantindo a actualização da biblioteca. Na data connection Web Service Update, é necessário definir qual a operação que iremos utilizar no web service, neste caso é a UpdateInq, cujo código se encontra no Anexo E.1.3 - Código-fonte C# do Web Service, e esta operação espera receber em cada variável do método, os valores respectivos do datasource que enviamos conforme o quadro abaixo:

Parâmetro (web service)	Tipo	Elemento (form)
Tns:idForm	Int	/my:statusReport/my:group37/my:p1
Tns:codPais	String	/my:statusReport/my:group37/my:p2
Tns:codEscola	Int	/my:statusReport/my:group37/my:p3
Tns:nomeEscola	String	/my:statusReport/my:group37/my:p4
Tns:ruaEscola	String	/my:statusReport/my:group37/my:p5
Tns:cidadeEscola	String	/my:statusReport/my:group37/my:p6
Tns:regiaoEscola	String	/my:statusReport/my:group37/my:p7

Tabela 14 - Passagem de parâmetros para o web service Update: método updateInq

Tal como na inserção, os dados do formulário, estão deste modo associados a cada parâmetro do web service, e a partir daqui toda a funcionalidade é realizada por parte do web service que recebe os parâmetros e os associa a uma string SQL e os realiza um update directamente na Base de dados. Assim, a biblioteca recebeu e actualizou os dados enviados na primeira acção da Rule, e o web service recebeu os parâmetros e inseriu-os na base de dados SQL, e isto de um modo totalmente transparente para o utilizador.

#### 5.4.3.3 Web Service - Operação de Delete na base de dados

A operação de Delete dos dados na Base de dados é muito semelhante á operação de Insert, e a sua grande diferença está no método de Delete do web service e na possibilidade de efectuar igualmente a remoção na biblioteca de formulários.

O formulário é praticamente idêntico, exceptuando a criação de um botão no formulário a que ficará associada às Rules de Delete. Este botão necessita ser criado pois não é possível associar as duas acções a uma operação de Delete num formulário web. Deste modo, para realizar uma remoção, teremos também de criar uma Rule de nome Delete. Esta Rule terá duas acções, a acção de remoção para a biblioteca de formulários terá o nome de Sharepoint Library Delete e será idêntica á do capítulo 5.4.1 – O armazenamento de dados em Listas Sharepoint e uma outra com o nome de Web Service Delete enviará o formulário para um web service com a operação de DeleteInqueritoByID conforme as tabelas seguintes:

Nome	Sharepoint Library Delete
Tipo	Submit Data (Apenas submete dados)
Lista ou biblioteca Sharepoint	<a href="http://inquerito:2222/munsi/MUNSIcrianca/">http://inquerito:2222/munsi/MUNSIcrianca/</a>
Nome do ficheiro .XML a criar	Com base no valor do datasource codCrianca (que identifica univocamente cada formulário)
Permitir overwrite	Se for ficheiro pré-existente

**Tabela 15 - Características de Sharepoint Library Delete**

Nome	Web Service Delete
Tipo	Submit Data (Apenas submete dados)
Lista ou biblioteca Sharepoint	<a href="http://inquerito:8090/service1.asmx">http://inquerito:8090/service1.asmx</a>
ID	Com base no valor do datasource codCrianca (que identifica univocamente cada formulário)
Operação	DeleteInqueritoByID

**Tabela 16 - Características do Web Service Update**

Na data connection Web Service Delete, é necessário definir qual a operação que iremos utilizar no web service, neste caso é a DeleteInqueritoByID, cujo código se encontra no Anexo E.1.3 - Código-fonte C# do , e esta operação espera receber em uma variável que identifica univocamente o formulário a remover, neste caso o valor idForm conforme o quadro abaixo:

Parâmetro (web service)	Tipo	Elemento (form)
Tns:idForm	Int	/my:statusReport/my:group37/my:p1

**Tabela 17 - Passagem de parâmetros para o web service Update: método updateInq**

Tal como na inserção, o dado do formulário que identifica o formulário, está associado ao parâmetro do web service, e a partir daqui toda a funcionalidade é realizada por parte do web service que recebe o parâmetro e o associa a uma string SQL que efectua um Delete directamente na Base de dados.

Assim, a biblioteca recebeu e actualizou os dados enviados na primeira acção da Rule, e o web service recebeu os parâmetros e removeu a entrada identificada na base de dados SQL, e isto de um modo totalmente transparente para o utilizador.

#### **5.4.4 Utilizar Workflows na integração de dados**

Esta é uma hipótese de solução, mas não foi explorada aprofundadamente devido á prevalência das duas soluções anteriores.

Conceptualmente esta solução permitiria através de condições, estados e acções na Lista de Sharepoint, realizar todas as tarefas de acesso e manipulação de dados numa base de dados SQL, que neste momento estão a ser realizadas pelo web Service. Esta opção é possível, sendo necessário realizar desenvolvimento, adaptar e actualizar a plataforma.

Ao inserir ou actualizar um formulário, seria automaticamente accionado um workflow que teria a capacidade de efectuar a ligação com um Provider SQLClient, ligando a uma datasource definida para uma base de dados, e utilizar os valores lidos da lista como datasource para os inserir ou actualizar por SQL numa tabela na Base de Dados.

Em caso de alterações no formulário, bastaria efectuar as alterações necessárias no workflow e se necessário, na base de dados, para que as alterações se reflectissem de imediato

Não implica isto que o Web Service não seja uma solução adequada às necessidades, muito pelo contrário, mas no âmbito de existir uma possibilidade de um administrador poder realizar alterações rapidamente e sem necessidade de desenvolvimento e alterações no código do web service, esta seria uma opção interessante, pois é um processo totalmente administrativo na plataforma.

#### **5.5 O Interface no processo de recolha de dados e partilha de informação**

O processo que envolve a utilização da solução é aqui apresentada dando primeiro a perspectiva do utilizador, relativamente às diferentes opções e passos que necessita realizar para submeter um formulário correctamente, em segundo lugar, o processo relativo ao administrador como responsável pela concentração dos dados preparando-os para serem tratados, em que são analisadas as diferentes opções possíveis de integração, utilizando o Microsoft Access, Excel ou ainda através da uma base de dados SQL.

### 5.5.1 Recolha de dados – Interface Utilizador

De modo a que todos os utilizadores pudessem utilizar a plataforma de um modo independente, toda o acesso é via Web através de um browser, onde o utilizador acede ao endereço <http://inquerito.uatla.pt>, e com do nome de utilizador e palavra passe que lhe foi atribuído anteriormente pela administração pode ter acesso á plataforma conforme a imagem seguinte:



Figura 16 - Acesso á Home page

Foram definidos e associados utilizadores a roles, roles esses que definem as permissões a cada site, pelo que apenas os utilizadores autenticados e com permissões poderão aceder aos seus sites.

Dentro dos seus sites, terão acesso á sua área de trabalho e à biblioteca de formulários que utilizarão nos inquéritos.

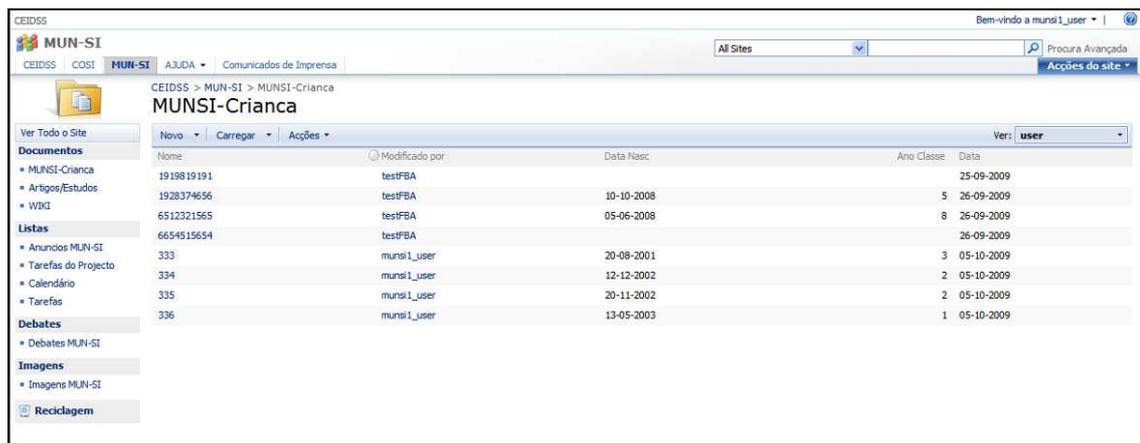


Figura 17 - Vista de biblioteca de formulários

A partir desta biblioteca de formulários, um utilizador pode aceder a várias funcionalidades tal como criar, editar ou remover um formulário. Pode ainda fazer *upload* de um ou vários formulários através do browser, criar vários tipos de vistas pessoais sobre a Lista, de modo a que possa visualizar apenas a informação que pretende analisar. Ao clicar na opção “Novo” irá carregar um formulário em branco pronto a ser preenchido.

**InqueritoMun-SI Crianca**

Questionário da Criança

MUN-SI

Universidade Atlântica

Código da Criança

**Identificação da Criança**

1. Qual o teu nome? Qual o teu apelido?

2. Sexo da Criança?  Rapaz  Rapariga

3. Data de nascimento da Criança? ex: dia-Mês-Ano

4. Localização da residência da Criança?

Código Postal - Concelho

5. Freguesia

6. Em que ano/classe estás?

7. Código da Escola

8. Nome da Escola

9. Morada da Escola

Rua, nº, localidade

10. Morada da Escola

Código Postal - Concelho

Figura 18 – Formulário no browser

Ao terminar de preencher os dados, tem a opção de submeter o formulário directamente para a biblioteca de documentos, ou guardar o documento com um nome de ficheiro específico. Ao enviar o formulário, se existir algum erro a nível de dados em falta, como por exemplo o do código de criança que é um campo que nunca pode ser nulo, um formato de data errada ou alguma inconsistência que vá contra as regras definidas na Business Logic do formulário, o utilizador não o poderá enviar enquanto não corrigir os erros. Se não existirem erros, o formulário será enviado, sendo mostrada uma mensagem de sucesso no envio e o browser é redireccionado de volta para a biblioteca, onde está já na lista o formulário criado.

Para editar um formulário já criado, basta que o utilizador clique na lista com a referência desejada e o formulário será carregado com os dados preenchidos por defeito. Poderá efectuar as alterações necessárias e voltar a submeter o formulário. Se não existirem erros, o formulário será enviado, sendo mostrada uma mensagem de sucesso no envio e o browser é redireccionado de volta para a biblioteca, onde está já na lista o formulário alterado.

Deste modo, o que é disponibilizado ao utilizador é toda a estrutura necessária para a partir de qualquer computador com acesso á Internet poder realizar o seu trabalho, criando os formulários de um modo simples, seguro e uniforme a todos os utilizadores.

### **5.5.2 Análise dos dados recolhidos – Interface Administração**

Um administrador terá acesso a funcionalidades mais avançadas, como abrir a biblioteca como uma folha Excel e editar dados directamente no browser. Esta opção é interessante para editar dados em bloco. Pode ainda ter acesso a uma vista de explorador, onde tem acesso aos ficheiros .XML de cada um dos formulários. Pode ainda exportar todos os dados para uma folha de cálculo ou ligar todos os dados ao Microsoft Access.

A integração da plataforma Sharepoint com o Microsoft Office é total e sem necessidade de ferramentas adicionais, o administrador pode obter todos os dados para análise em duas ferramentas, o Microsoft Excel ou o Microsoft Access.

### 5.5.2.1 Análise dos dados com Microsoft Excel 2007

Para um administrador proceder á análise dos dados, pode utilizar o Microsoft Excel a partir da própria biblioteca de formulários, onde pode editar directamente na biblioteca com a opção Editar na Folha de Dados ou abrindo directamente na aplicação cliente Microsoft Excel com a opção “Exportar para folha de cálculo”.

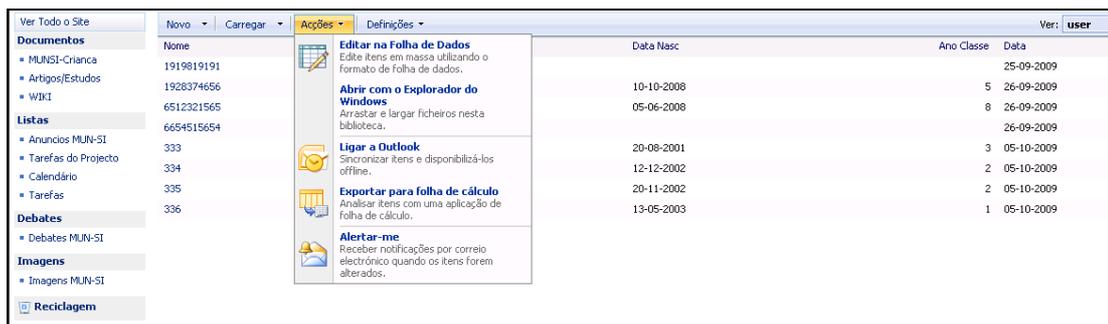


Figura 19 - Vista de funcionalidades

Para efectuar o carregamento dos dados, o Sharepoint não exporta directamente o conteúdo dos dados da lista, pois apenas necessita de disponibilizar um ficheiro com toda a configuração necessária para a ligação do cliente Excel á plataforma.

Essa ligação é um ficheiro de nome owssvr.iqy que é um ficheiro de query Web do Microsoft Office Excel e o que este ficheiro permite é que o Excel abra com uma ligação directa á vista que exportamos e obtenha daí todos os dados.

É aqui que estão definidas a string de ligação á biblioteca com um Microsoft.Office.List.OLEDB.2.0, o Command Text para aceder á lista da biblioteca de onde exportamos o ficheiro .IQY e finalmente a autenticação Windows, visto ser um dos requisitos que a análise dos dados seja feita pela administração da plataforma, e garantindo deste modo que seja impossível qualquer funcionalidade de exportação fora do Domínio onde está a plataforma.

Essa ligação possui as seguintes características:

Connection string	Provider=Microsoft.Office.List.OLEDB.2.0; Data Source="";ApplicationName=Excel; Version=12.0.0.0
Command text	<pre>&lt;LIST&gt;   &lt;VIEWGUID&gt;     {E74DB095-0A6C-4DB1-B9BF-83DFB2C3CECB}   &lt;/VIEWGUID&gt;   &lt;LISTNAME&gt;     {8A1F5C50-2153-4BBB-997A-F5F34AB06AA5}   &lt;/LISTNAME&gt;   &lt;LISTWEB&gt;http://inquerito:2222/munsi/_vti_bin&lt;/LISTWEB&gt;   &lt;LISTSUBWEB&gt;&lt;/LISTSUBWEB&gt;   &lt;ROOTFOLDER&gt;/munsi/MUNSIcrianca&lt;/ROOTFOLDER&gt; &lt;/LIST&gt;</pre>
Autenticação	Windows

Tabela 18 - Exemplo de web query para Excel

Deste modo, é possível obter numa folha de cálculo todos os dados da lista de SharePoint, e proceder á sua análise directamente no Excel, conforme a figura abaixo com toda a informação da lista, incluindo o ficheiro .XML.

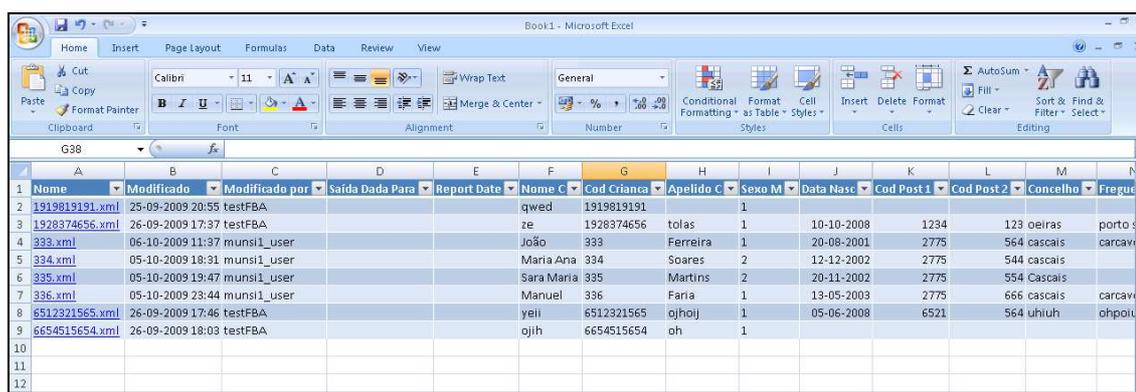


Figura 20 – Lista Exportada para Excel

A partir da ferramenta SPSS este ficheiro poderá ser importado directamente, onde toda a análise estatística será realizada.

### 5.5.2.2 Análise dos dados com Microsoft Access 2007

Outra forma de obter os dados é utilizando o Microsoft Access 2007, que tal como a opção anterior, está totalmente integrado com a plataforma MOSS.

Com o Microsoft Access, apenas é necessário criar uma nova tabela e através da opção de External Data podemos aceder a uma datashource com base numa Lista de Sharepoint da biblioteca que se pretender analisar. Especificando o endereço do site e qual a biblioteca de formulários que desejamos importar.

Como opções de fonte e destino de dados, podemos definir se desejamos criar uma Linked Table, onde é criada uma tabela que mantém um link directamente com a lista e que pode ser acesso a uma vista dos dados em tempo real ou então ter acesso a uma cópia integral dos dados até aquele momento para trabalhar off-line.

Os dados da lista serão todos importados directamente para a tabela criada e tal como na opção Excel todos os dados são importados, incluindo os ficheiros .XML tal como podemos verificar na imagem seguinte:

ID	Nome	Titulo	Nome C	Cod Crianca	Apelido C	Sexo M	Data Nasc	Cod Post 1	Cod Post 2	Concelho	Fregues
1	1919819191.xml	1919819191.xml	qwed	1919819191		1					
2	1928374656.xml	1928374656.xml	ze	1928374656	tolas	1	10-10-2008	1234	123	oeiras	porto salv
3	6512321565.xml	6512321565.xml	yeii	6512321565	ojhoij	1	05-06-2008	6521	564	uhuh	ohpoiu
4	6654515654.xml	6654515654.xml	ojih	6654515654	oh	1					
5	333.xml	333.xml	João	333	Ferreira	1	20-08-2001	2775	564	cascais	carcavelos
6	334.xml	334.xml	Marja Ana	334	Soares	2	12-12-2002	2775	544	cascais	
7	335.xml	335.xml	Sara Maria	335	Martins	2	20-11-2002	2775	554	Cascais	
8	336.xml	336.xml	Manuel	336	Faria	1	13-05-2003	2775	666	cascais	carcavelos

Figura 21 - Vista Tabela em Access

Visto neste caso termos uma tabela de uma base de dados, na importação dos dados a estrutura é garantida, pois a partir das definições do tipo de dados e nome dos controlos do

formulário durante a sua criação com o InfoPath, as mesmas características vão se replicar na tabela de Access. Deste modo, na tabela criada, tanto o Field Name como os diferentes data types são idênticos aos definidos na Lista da biblioteca de formulários.

A partir da ferramenta SPSS este ficheiro poderá ser importado directamente, onde toda a análise estatística será realizada.

### **5.5.2.3 Análise dos dados com Microsoft SQL Server 2005**

Esta forma de obter os dados está apenas disponível para a opção de recolha de dados com a funcionalidade de um Web Service. O web service criado foi instalado e configurado no url: <http://webservice:8090> e efectua o carregamento e actualizações de todos os dados na tabela da base de dados, tal como descrito no capítulo 5.4.3 - Utilizar Web Services na integração de dados

A base de dados de inquéritos está instalada na instância CEIDSS do SQL Server 2005 da plataforma tendo sido criada uma tabela com as características descritas no Anexo E.1- Script SQL. Esta Base de Dados é apenas acessível a partir do próprio servidor ou de um sistema dentro do domínio

Neste caso, apenas os dados são enviados para a tabela, ao contrário das outras duas opções descritas nos capítulos anteriores que recebiam além dos dados, o ficheiro .XML do formulário. As características da tabela são adaptadas aos dados recolhidos e a partir desta base de dados, é possível efectuar os diferentes tratamentos e análises que o administrador necessite. Tal como nas outras opções, com a ferramenta SPSS este ficheiro poderá ser importado directamente, onde toda a análise estatística será realizada.

## 6 Trabalhos Futuros

No âmbito de utilização de uma ferramenta de trabalho colaborativo, a plataforma pode atingir um nível de funcionalidades muito abrangente sem grandes exigências técnicas sendo por isso facilmente escalável.

Para a utilização da plataforma como uma ferramenta de trabalho completa, deverão ser disponibilizadas todas as ferramentas que se considerarem necessárias. A plataforma possui de raiz várias opções que podem ser implementadas em cada site de Investigação, que podem facilitar a utilização e aumentar bastante o nível de produtividade, pois são ferramentas online que permitem que um utilizador possa ter a gestão do seu trabalho totalmente integrada na Plataforma, de um modo centralizado e sem necessidade de ferramentas adicionais.

Além das funcionalidades já implementadas a nível de formulários, é possível adicionar outras bibliotecas de documentos, onde os utilizadores podem aceder ou criar documentos Microsoft Word, Excel, PowerPoint, Pdf, ou imagens, disponibilizando deste modo um repositório de informação para onde todos podem contribuir.

Podem ser também adicionadas funcionalidades de Listas de tarefas e Calendários integrados com o Microsoft Outlook, onde podem ser geridas tarefas atribuídas a cada investigador, uma funcionalidade interessante para coordenar os trabalhos a realizar.

Outra funcionalidade bastante interessante como um trabalho futuro, é a possibilidade de ser disponibilizada pela plataforma a funcionalidade MySite, que é um site totalmente independente do site Inquéritos, mas que é gerido pela mesma plataforma de gestão. Este site é criado especificamente para o utilizador, de um modo totalmente autónomo, com base no utilizador da plataforma. Deste modo, um utilizador normal da plataforma pode ter acesso a um site pessoal. e que é totalmente personalizável pelo utilizador, tendo deste modo acesso a ferramentas e funcionalidades escolhidas pelo próprio, como *blogs*, *web parts*, bibliotecas, calendários, fóruns, sites de debate.

Pode inclusive ter acesso a funcionalidades que não estão disponíveis no site Inquéritos ou criá-las de raiz para sua utilização própria. Este MySite, tal como no site Inquéritos, tem

funcionalidades que são possíveis de ser totalmente integradas com as aplicações cliente Microsoft Office ou ainda com a própria plataforma Inquéritos. Assim, um utilizador pode ter todas as funcionalidades, tanto do site principal como as funcionalidades “pessoais”.

Em resumo, propõem-se que o utilizador se torne totalmente autónomo na utilização de várias ferramentas totalmente orientadas para as suas funções, ferramentas essas que visam facilitar e centralizar o seu trabalho de um modo simples, seguro e sempre disponível.

## **7 Conclusão**

Não podemos negar que se caminha cada vez mais para uma uniformização e centralização de processos seja numa organização ou num departamento de investigação. A evolução e aparecimento de novos serviços e soluções de base tecnológica é hoje em dia em tão grande número que é necessário encontrar um equilíbrio entre essa inovação e as necessidades da organização de modo a que seja possível acompanhar todas estas mudanças criando valor para a organização.

Uma plataforma de trabalho colaborativo é uma forma apelativa de inovação tecnológica que se apresenta de forma amigável, uniforme e coerente para todas as partes envolvidas num processo que normalmente é reconhecido como de rápido retorno.

Com uma plataforma como o MOSS, é possível disponibilizar várias funcionalidades tal como bibliotecas de documentos, Listas, Tarefas ou Formulários e com total integração com aplicações cliente como o Microsoft Office bastante familiares ao utilizador.

Uma plataforma com as características do MOSS, permite que qualquer utilizador, possa ter acesso a uma estrutura organizada e centralizada, para onde pode contribuir, sem ter necessidade de aplicações específicas ou equipamentos dedicados, utilizando apenas um browser e acesso á Internet, com todas as funcionalidades necessárias para realizar o seu trabalho.

No âmbito deste trabalho, no contexto de uma plataforma colaborativa a ser utilizada por um departamento de Investigação, as funcionalidades e possibilidades oferecidas adequam-se na totalidade às necessidades encontradas tanto a nível de recolha como de análise dos dados. A recolha é possível através de qualquer equipamento com um browser e acesso á internet ou através da aplicação cliente InfoPath, e independentemente da escolha feita, os dados recolhidos mantêm as mesmas características e qualidade.

Do ponto de vista da análise dos dados recolhidos, a capacidade de integração da plataforma com diferentes ferramentas, tal como o Microsoft Office Excel ou o Microsoft Office Access, permite que sejam realizados, em tempo real, a análise ou tratamento dos dados

recolhidos, ou ainda, que sejam tratados por ferramentas de análise estatística não Microsoft como o SPSS.

Depois de adoptada a solução de integração com as listas do SharePoint, solução nativa, foi estudada uma solução alternativa de integração, utilizando Web Services que permite a automatização do processo de passagem dos dados registados nos formulários para uma base de dados SQL Server. Qualquer das soluções foi testada com sucesso e sem qualquer tipo de perda de qualidade dos dados recolhidos e sem impacto no processo do utilizador.

Relativamente à opção pela solução nativa, interessa salientar que não foi por de algum modo ser superior à outra, mas apenas pelo motivo de ser muito mais simples a manutenção e escalabilidade, tanto a nível do formulário em si, como na recolha de dados. A fácil manutenção e escalabilidade da solução eram requisitos iniciais do projecto. A opção de recolha utilizando um Web Service e uma base de dados, implica em caso de qualquer tipo de alteração a nível dos dados no formulário, que essa alteração se reflecta tanto no código do Web Service como na base de dados SQL. Esta situação obriga a desenvolvimentos adicionais, com os respectivos tempos de desenvolvimento, teste e passagem a produção, todos estes com um custo adicional para o processo de investigação.

Deste modo, tanto a nível da plataforma criada, dos serviços e funcionalidades de trabalho centralizado e solução de disponibilização de formulários e recolha de dados, é da opinião do autor que toda a plataforma pode ser uma mais-valia para o departamento de investigação tanto nível de processos e procedimentos do inquiridor ou do investigador, adequando-se a todas as necessidades encontradas.

Esta pode ser sem dúvida uma ferramenta bastante útil no trabalho de investigação desenvolvido pelo departamento.

## 8 Bibliografia

- Roberts, S. e Green, H (2007). *Designing Forms for Microsoft Office InfoPath and Forms Services 2007* (1º Ed.). Boston: Pearson Education.
- Hoffman, K e Foster, R. (2007). *Microsoft SharePoint 2007 Development Unleashed*. Ed. (2ª Ed.). Indianapolis: Sams Publishing.
- Powers, S. (data desconhecida). *Forrester report - Web Content Management and Open Source, Answers to Frequently Asked Questions*, Disponível *on-line* em: <http://acquia.com/node/274>. Último acesso em 15-06-2009.
- Krüger, Jörg (03/2004). *Open Source vs. commercial CMS*. Disponível *on-line* em: [http://www.contentmanager.net/magazine/article\\_257\\_open\\_source\\_commercial cms.html](http://www.contentmanager.net/magazine/article_257_open_source_commercial cms.html). Último acesso em 25-06-2009.
- Markun, P, (07/2009). *Content Management System*. Disponível *on-line* em: <http://www.websitemagazine.com/content/blogs/posts/pages/choosing-the-best-web-content-management-system.aspx>. Último acesso em 12-08-2009.
- Desconhecido. (08/2009). *Limitation of Using MS InfoPath 2007*. Disponível *on-line* em: <http://webdesignmarketing.iblogger.org/2009/08/27/limitation-in-infopath/>. Último acesso em 25-08-2009.
- Scott (11/2005). *Membership and Role Providers in ASP.NET 2.0*. Disponível *on-line* em: <http://odetocode.com/Articles/428.aspx>. Último acesso em 30-08-2009.
- Braga, A (data desconhecida). *A Gestão da Informação*. Disponível *on-line* em: [http://www.ipv.pt/millennium/19\\_arq1.htm](http://www.ipv.pt/millennium/19_arq1.htm) Último acesso em 03-06-2009.
- Desconhecido, (data desconhecida). *Microsoft Office Forms Server 2007*. Disponível *on-line* em: <http://sharepoint.microsoft.com/Pages/Default.aspx>. Último acesso em 03-06-2009.
- Connel, A. (12/2006). *Configuring an Office SharePoint Server 2007 Publishing Site with Dual Authentication Providers and Anonymous Access*. Disponível *on-line* em: <http://www.andrewconnell.com/blog/articles/HowToConfigPublishingSiteWithDualAuthProvidersAndAnonAccess.aspx>. Último acesso em 03-09-2009.





Licenciatura em Gestão de Sistemas e Tecnologias de Informação

## **Anexos do Projecto**

### **Formulários on-line em Plataforma Colaborativa**

Elaborado por Rui Miguel Charneca

Aluno nº 2000 586



Orientador: Professora Filipa Taborda Ferreira

Barcarena

Junho 2009

Universidade Atlântica

Licenciatura em Gestão de Sistemas e Tecnologias de Informação

## **Anexos do Projecto**

### **Formulários on-line em Plataforma Colaborativa**

Projecto Final de Licenciatura

Elaborado por Rui Miguel Charneca



Aluno nº 2000 00 586

Orientadora: Professora Filipa Taborda Ferreira

Barcarena

Novembro 2009



## ANEXOS

Anexo A.	Arquitectura Sharepoint.....	1
A.1	Arquitectura WSS.....	1
A.2	Arquitectura MOSS.....	2
A.2.1	Arquitectura Lógica.....	4
A.2.2	Recursos Necessários.....	5
A.2.3	Arquitectura Física.....	6
A.2.4	Arquitectura administrativa.....	10
A.2.5	Segurança no Sharepoint.....	10
A.2.6	Anonymous Authentication.....	13
A.2.7	Windows Authentication.....	13
A.2.8	Forms Based Authentication.....	13
A.2.9	Permissões Sharepoint.....	14
A.2.10	Hierarquia e Herança.....	16
A.2.11	Utilizadores e grupos de Sharepoint.....	17
A.2.12	Tipos de conteúdos e funcionalidades.....	17
A.2.13	Site Content Type.....	18
A.2.14	Bibliotecas e Listas de Sharepoint.....	18
A.2.15	Web Parts.....	19
A.2.16	Workflows.....	20
Anexo B.	Web Service.....	21
B.1	SOAP.....	22

Anexo C.	Form Server e Microsoft Office InfoPath 2007 .....	23
C.1	Microsoft Office InfoPath 2007 .....	23
C.2	Integração InfoPath e SQL Server .....	27
C.3	Forms Services.....	29
C.3.1	Opção Form Services para Web browser .....	29
C.3.2	Requisitos Mínimos para o Form Services .....	30
C.3.3	Browsers Suportados .....	31
Anexo D.	Plataforma a instalar .....	32
Anexo E.	Código e Scripts Utilizados .....	34
E.1	Script SQL .....	34
E.1.1	Script SQL da Base de Dados.....	34
E.1.2	Script SQL da Tabela Inquerito.....	36
E.1.3	Código-fonte C# do Web Service.....	37
Anexo F.	Manual de utilizador .....	41
Anexo G.	Visão da Estrutura Implementada .....	42

## Anexo A. Arquitectura SharePoint

O Sharepoint encontra-se na sua terceira *release* e inclui o Windows SharePoint Services (WSS) versão 3.0 e o Microsoft Office SharePoint Server (MOSS) 2007. O WSS v3 é um *add-on* gratuito no Windows 2003 Server e tem como base o SQL Server, Windows 2003 Server e o ASP.NET 2.0.

O MOSS está disponível em várias edições (Standard vs. Enterprise) e opções (Excel Services, Content Management, etc.), e corre com base no WSS. A arquitectura do MOSS é bastante semelhante á do Windows SharePoint Services (WSS), mas disponibiliza bastantes mais serviços e aplicações. A arquitectura da plataforma baseia-se na Gestão de utilizadores e segurança, IIS e ASP.NET

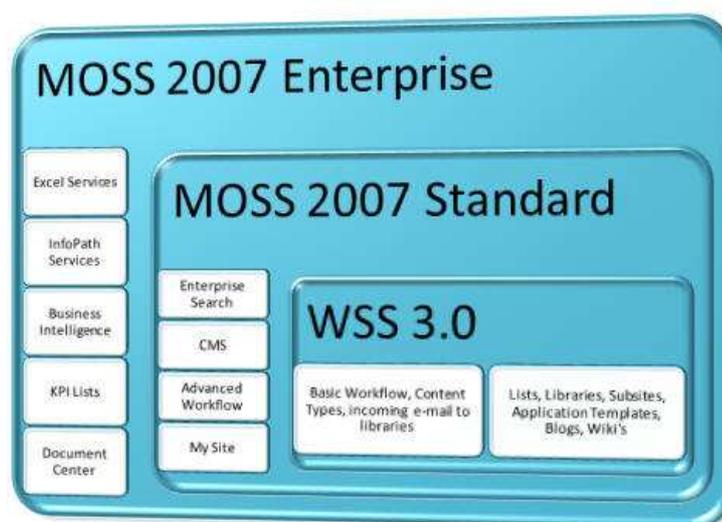


Figura 22 - Evolução da Arquitectura Sharepoint

### A.1 Arquitectura WSS

O WSS é a plataforma principal do Sharepoint Services, composta pelo Front Web Server, Servidor de procura e indexação e um servidor de Base de dados.

O WSS é essencialmente baseado numa aplicação web ASP.NET, que estende um website no IIS, processando HTML através de páginas .aspx, APIs .NET e Web services XML. Processa e executa business logic combinando .NET e assemblies Sharepoint. Todos os

dados são guardados numa base de dados SQL. O Sharepoint apresenta a informação aos utilizadores em HTML através de um browser compatível, como o Internet Explorer. Estendendo um Web site no IIS com o WSS obtemos uma Web Application que utiliza um HttpModule e um HttpHandler para redireccionar o trafico recebido para a business logic do Sharepoint, permitindo que uma Web Application em Sharepoint possa existir entre outras Web applications no IIS.

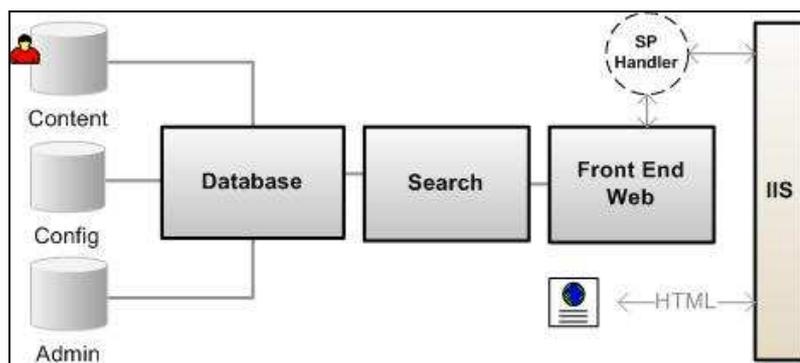


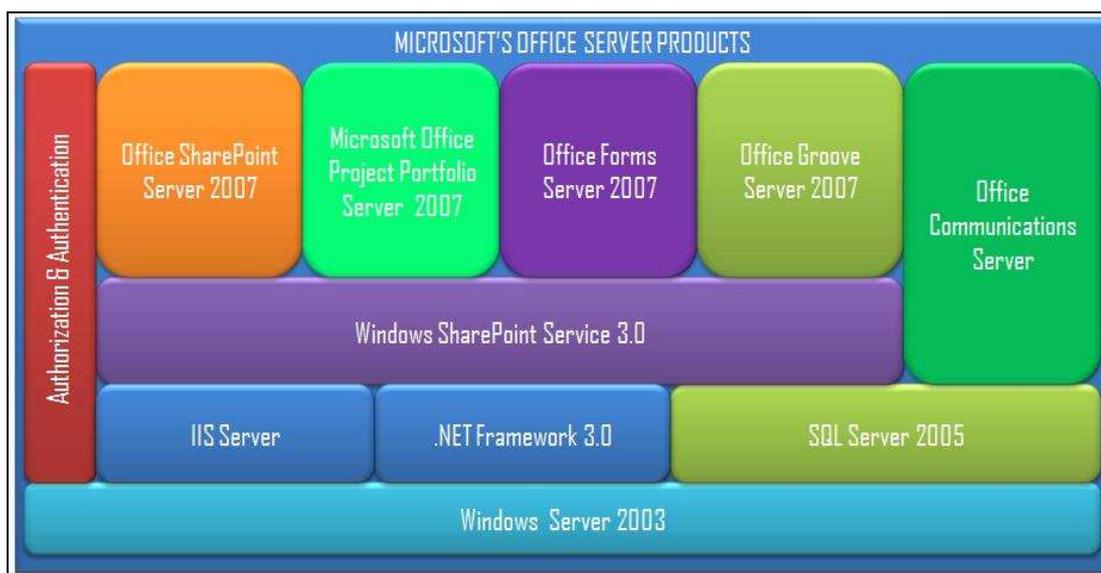
Figura 23 – Arquitectura WSS

## A.2 Arquitectura MOSS

Embora o MOSS tenha como base a plataforma WSS, possui bastantes aplicações e opções, tal como Gestão avançada de conteúdos e publicação de sites, procura de conteúdos em bases de dados Externas e bastantes site templates<sup>5</sup> e workspaces<sup>6</sup>.

<sup>5</sup> Templates – Sites pré-definidos out-of-the-box, relativos a vários temas, tal como gestão de contactos ou um Call Center, que podem ser simplesmente adicionados, possuindo um ambiente gráfico aceitável e sem qualquer tipo de desenvolvimento necessário pelo utilizador.

<sup>6</sup> WorkSpace – Website que disponibiliza aos membros de uma equipa de trabalho as ferramentas e aplicações necessárias para um trabalho colaborativo. Inclui listas, documentos, webparts.



**Figura 24 – Arquitectura MOSS 2007**

O MOSS possui ainda uma aplicação de nome Shared Services, que podem ser configurados num servidor e partilhados por diferentes portais MOSS e WSS, contendo aplicações corporativas com pesquisa, indexação, perfis de utilizadores, gestão de conteúdos, MySites, Business Data Catalogue, Form Services, Excel Services, Job Scheduling e Relatórios de utilização. Outra diferença estrutural do WSS é que a base de dados de pesquisa é criada para cada SSP (*Shared Services Provider*) na Farm. Esta base de dados contém informação de pesquisa como propriedades de crawl<sup>7</sup>, de documentos e de propagação<sup>8</sup>.

<sup>7</sup> Crawl – definição de regras que delimitam os conteúdos a serem pesquisados tal como pastas numa rede.

<sup>8</sup> Propagação – definição de processo de indexação de modo a balancear recursos de pesquisa intensiva.



Figura 25 - Pilares do MOSS

### A.2.1 Arquitectura Lógica

A arquitectura lógica do MOSS 2007 disponibiliza uma Framework com componentes lógicos onde se incluem Farms de servidores, Web applications, Site Collections, sites Web e features. No topo da hierarquia temos os servidores da Farm. Uma Farm de servidores é um agrupamento lógico de application servers que partilham a mesma base de dados de configuração. Esta base de dados contém informação sobre os roles de cada servidor na Farm, onde por exemplo, um servidor pode ser um Web Front End ou uma base de dados, ou um servidor de mail, etc.

O segundo nível da hierarquia são as Web applications que disponibilizam funcionalidades de Web Server. Quando se cria uma Web application no MOSS 2007 é criado um Web site correspondente no IIS. Este site contém as definições que serão aplicadas a um grupo de site collections sobre aquela Web application.

O terceiro nível é o das Site collections que determinam as definições e contexto para agrupar um conjunto de sites e subsites.

O nível seguinte é o site, que é um agrupamento de dados numa lista, biblioteca ou Web part. O site disponibiliza os serviços num link único, a listas de utilizadores autenticados com permissões.

No fim desta hierarquia temos as features que são elementos customizados, que disponibilizam serviços específicos e podem ser dados, metadata ou funcionalidades e podem ser utilizados em sites ou site collections.

### A.2.2 Recursos Necessários

Os recursos mínimos necessários á instalação da Plataforma MOSS num servidor são os da seguinte tabela:

<b>Componente</b>	<b>Requisito</b>
Computador e processador	Servidor com velocidade do processador de 2,5 GHz ou superior; processador duplo de 3 GHz ou superior recomendado
Memória	1 GB de RAM; 2 GB recomendado
Disco rígido	NTFS com 3 GB de espaço disponível no disco rígido
Sistema operativo	Microsoft Windows Server 2003 Standard Edition, Windows Server 2003 Enterprise Edition, Windows Server 2003 Datacenter Edition ou Windows Server 2003 Web Edition ou posterior e service packs. Microsoft Windows SharePoint Services 3.0 e IIS
Outros	Velocidade de ligação de 100 Mbps necessária para implementação de Farm.
Browser	Internet Explorer 5 ou superior, Firefox
Framework	Microsoft .NET Framework 3.0 e ASP.NET 2.0
Base de dados	Microsoft SQL Server 2005

**Tabela 19- Tabela de Requisitos**

Estes requisitos são identificativos, e são a recomendação do fabricante, devendo de acordo com cada implementação ser ajustados ao ambiente onde será implementado.

As possibilidades de integração da plataforma com outros sistemas são enormes e facilmente configuráveis, com inúmeras opções de configuração e de interligação de serviços, como por ex. Active Directory, Exchange Server, SQL Server, Reporting Services, etc., incluindo a integração a nível do utilizador com todos os produtos Microsoft Office disponíveis até ao momento.

### **A.2.3 Arquitectura Física**

A arquitectura lógica do MOSS 2007 deve estar sempre em equilíbrio com a arquitectura física onde estão os dados. De modo a configurar correctamente a estrutura física, é importante que se conheçam bem os componentes lógicos da arquitectura MOSS.

A arquitectura física do MOSS 2007 consiste num modelo de três camadas, com os servidores Web na frente, a seguir os Application Servers e finalmente o servidor de base de dados.

#### **1. Front End Server**

Gere os pedidos para a aplicação MOSS. Contem várias directorias virtuais que permitem features das aplicações tal como, gestão de páginas, templates, temas e Web parts.

#### **2. Application Server**

Permite serviços para gestão de perfis de pesquisa, indexação, serviços Excel, gestão de Audiências, Perfis de utilizadores, bases de dados, MySites, Business Data Catalogue e sincronização. Podem ser configurados servidores diferentes para diferentes serviços.

#### **3. Base de Dados**

Armazena e gere vários tipos de dados, tanto gerados pelas aplicações como pelos utilizadores, desde o conteúdo, configurações, Shared Services Provider, pesquisas, metadata, etc. A Base de dados é responsável pelo controlo de toda a Farm.

A topologia da arquitectura pode ser diversificada e devemos ter em consideração factores como:

- N° de utilizadores
- Tipo de autenticação
- Requisitos de performance cliente/Servidor
- SLAs (uptime, requisitos de alta disponibilidade)
- Recursos WAN
- Quantidade de dados GBs/TBs
- N° total de ficheiros e itens.

Estes factores devem ser cuidadosamente analisados e nunca menosprezada sob pena de tornar toda a plataforma inadequada ao fim a que se propõe.

As diferentes opções vão desde um Single Server, a múltiplos Server Farms, onde diferentes configurações podem ser utilizadas, dependendo das necessidades e requisitos da organização.



**Figura 26 - Estrutura Single Server**

Single Server – A nível de recursos, é o mínimo possível para uma solução Sharepoint, e envolve apenas um servidor, que disponibiliza todo os serviços de Web Front End, Applications e base de dados. É a opção com menos performance e eficácia.



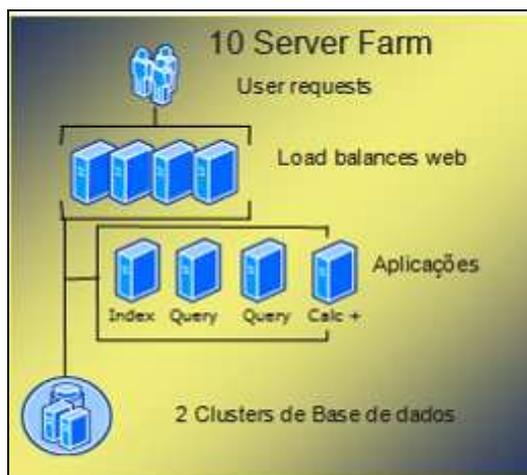
**Figura 27 Estrutura Small Farm**

Small Farm – Permite distribuir os serviços por 3 servidores, onde com 2 garantem os serviços Web e aplicações de modo a balancear a carga e um outro servidor dedicado apenas para a Base de dados. É uma opção que proporciona mais performance e disponibilidade mas ainda assim e pode ser implementada em portais corporativos de pequena a média dimensão.



**Figura 28 - Estrutura Média Server Farm**

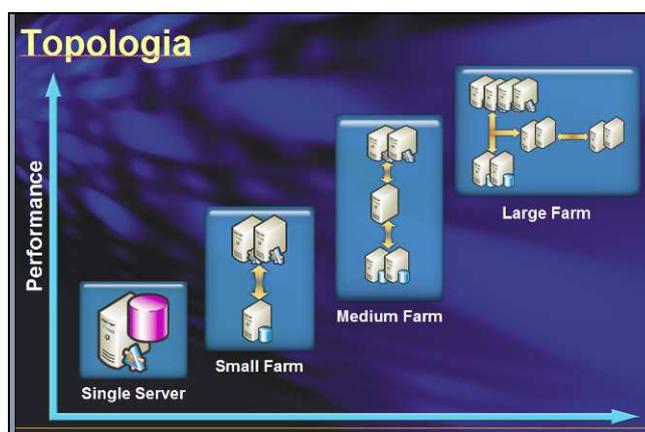
Média Server Farm – Permite distribuir os serviços por mais servidores, onde com 2 garantem os serviços Web, um servidor adicional para as aplicações e finalmente um cluster de servidores de Base de dados. Esta opção já permite uma performance e disponibilidade acima da média e pode ser adoptada a uma solução de médias a grandes dimensões.



**Figura 29 - Estrutura Large Server Farm**

Large Server Farm – Permite distribuir por mais servidores ainda, onde 4 garantem o serviço de balanceamento de carga nos front end Web Servers, existem 2 clusters de Base de dados e finalmente temos um servidores de aplicações para cada serviço como Indexação, dois servidores de pesquisa, Project Server ou outros. É indicada para serviços de alta disponibilidade, performance e redundância em grandes organizações com um número elevado de utilizadores, aplicações críticas para a organização onde não podem existir falhas de serviço e acesso Intranet e Internet.

É importante referir que a escolha da topologia tem influência directa na qualidade dos serviços que queremos disponibilizar, conforme o gráfico seguinte.



**Figura 30 – Topologia e performance**

Em conclusão, o planeamento da arquitectura física e lógica são passos importantíssimos na configuração da plataforma, onde deve ser feita uma análise cuidada entre a estrutura existente e a estrutura que se pretende implementar.

#### **A.2.4           Arquitectura administrativa**

A arquitectura administrativa contém três níveis de configuração, que são armazenados numa base de dados SQL.

##### **Nível 1**

Este nível armazena e gere os dados gerados por aplicações e utilizadores. Podem ser dados de configurações, de sites, metadata ou dados de indexação. Todos os membros de um Server Farm têm de usar o mesmo servidor de base de dados pois é o servidor de base de dados que armazena e gere a base de dados de configuração e é responsável pelo controlo das definições de toda a Farm.

##### **Nível 2**

Este nível inclui todas as features e funcionalidades administrativas para gestão dos shared services através da Farm. Por exemplo a configuração de pesquisa, indexação ou reports de utilização.

##### **Nível 3**

Este nível inclui várias opções de gestão específicas dos sites, tal como webparts, gestão de acessos e de conteúdo. A administração deste nível é por exemplo a criação de uma nova lista num site, configurar as permissões de acesso e alterar a hierarquia do site

#### **A.2.5           Segurança no Sharepoint**

A segurança no Sharepoint consiste em dois blocos: autenticação e permissões. O Sharepoint não administra por si só a autenticação, pois baseia-se no Windows, ASP.NET e IIS para o fazer. Por defeito, a opção *out-of-the-box* é Windows Authentication mas também

permite autenticação com FBA (*forms based authentication*) em SQL Server, a opção que será utilizada no projecto.

É possível utilizar os seguintes sistemas de gestão de identidade:

- a) Windows: Todos os relativos ao IIS e Windows authentication, incluindo Basic, Digest, Certificates, Windows NTLM e Kerberos. O Windows authentication permite que o IIS faça a autenticação do SharePoint.
- b) Forms ASP.NET: é um sistema de gestão não-Windows, que utiliza o sistema Microsoft ASP.NET form-based authentication. É possível utilizar vários tipos de gestão como LDAP ou SQL, sempre através de redireccionamento para uma página de login do utilizador.

Neste trabalho, utilizamos vários métodos de autenticação para aceder às aplicações Web do Sharepoint, visto não podermos utilizar a Active Directory da organização, a Zona Internet utilizará para os colaboradores que terão acesso aos formulários, o tipo Forms based Authentication com SQL Server, (Partner Site na imagem abaixo) onde serão introduzidos os utilizadores na Base de dados e definidas as permissões em grupos.

Para os utilizadores internos e de administração (que será um grupo muito restrito), será utilizado autenticação Windows, através de um utilizador criado no Servidor (Internal Site na imagem abaixo).

Para utilizar dois ou mais tipos de autenticação, é necessário estender as WebApplications e configurar Zonas adicionais, configurando um método de autenticação para cada uma das Zonas.

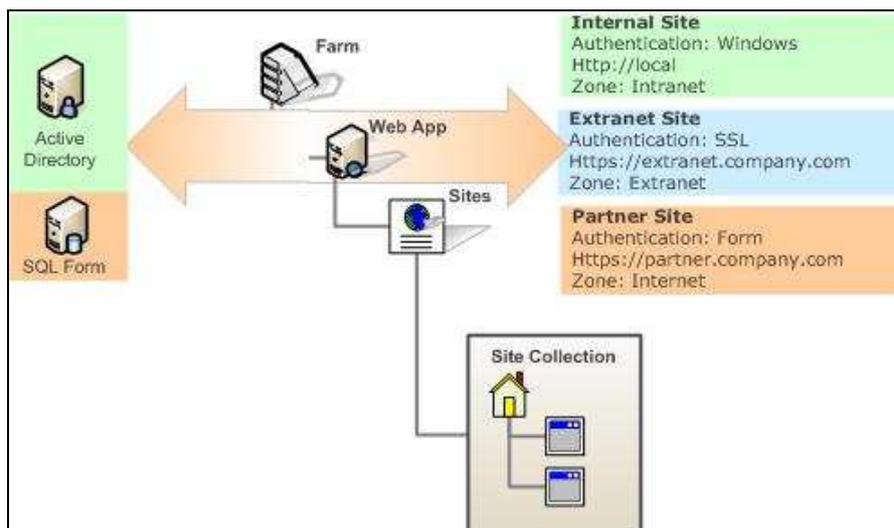


Figura 31 – Zonas e Métodos de autenticação do Sharepoint

Com a utilização de zonas distintas, permitimos que diferentes tipos de utilizadores possam aceder ao mesmo site, através de links diferentes, com autenticação distinta. Estes cenários são aplicados na distinção entre uma aplicação interna para colaboradores, ou uma aplicação acessível na internet por milhares de utilizadores.

As zonas existentes no Sharepoint são as seguintes:

- Custom – Zona de administradores com acesso local e autenticação Kerberos
- Intranet – zona de colaboradores utilizada para acesso interno por Windows authentication, normalmente com suporte de Active Directory
- Default – Zona de colaboradores remotos com autenticação Windows ou FBA com LDAP
- Extranet – zona de colaboradores remotos da organização, com autenticação FBA
- Internet – zona para clientes, normalmente com acesso anónimo para que permita a navegação inicial e posterior login por FBA com permissão para utilizadores anónimos

É ainda possível controlar os recursos atribuídos para cada uma das zonas de modo a balancear a plataforma distribuindo os recursos onde eles forem mais necessários

### **A.2.6 Anonymous Authentication**

Este tipo de autenticação permite o acesso a utilizadores que não tenham contas na plataforma ou servidor, como é o caso de utilizadores via Web de modo a que possam aceder à página de inicial. Este método não utiliza qualquer tipo de confirmação de identidade e permite um acesso livre aos recursos. Para um site Sharepoint permitir acesso anónimo é necessário primeiro configurar o IIS para permitir este tipo de acesso, e posteriormente configurar o próprio site no Sharepoint.

### **A.2.7 Windows Authentication**

Este método é mais seguro do que a autenticação básica devido ao facto de encriptar, utilizadores e passwords, numa interacção de múltiplas transacções entre cliente e servidor. Esta pode ser baseada em users criados no Servidor ou numa Active Directory. Este é o tipo de autenticação utilizado numa intranet corporativa, com autenticação com base na AD, onde a manutenção e gestão é muito mais facilitada.

### **A.2.8 Forms Based Authentication**

Forms based authentication utiliza um *ticket* de autenticação que é criado quando um utilizador se autentica no site, e acompanha o utilizador através da sua navegação pelo site. Normalmente este *ticket* é um *cookie*. Se um utilizador pedir acesso a uma página que necessita autenticação e o utilizador não se encontra autenticado, o utilizador é redireccionado para uma página de login, onde lhe são pedidas as credenciais de acesso, username e password. Estas credenciais são assim passadas para um servidor e validadas numa base de dados de utilizadores, que pode ser uma base de dados SQL ou uma Active Directory. Se essas credenciais forem válidas, a página cria um *cookie* que será utilizado durante essa sessão.

Neste método, podemos variar o sistema de gestão de autenticação, onde permite utilizar SQL Server, XML, Windows Live IDs, etc. Para o utilizador é apenas um formulário de autenticação que pede o *username* e a *password*, para a administração é a possibilidade de administrar utilizadores e grupos num sistema independente de uma Active Directory.

### A.2.9 Permissões Sharepoint

Depois de um utilizador ter sido devidamente autenticado para aceder, o Sharepoint pode determinar a que objectos o utilizador pode aceder e que tipo de operações pode realizar. Podemos assim distinguir quatro conceitos importantes no Sharepoint:

- Permissão
- Níveis de Permissão
- Securable Objects
- Grupos de Sharepoint

Permissões são os direitos que um utilizador tem de executar acções específicas, como navegar por páginas, editar itens, criar Subsites ou conteúdos. Deste modo, grupos de utilizadores podem ser facilmente geridos relativamente ao que podem ou não fazer.

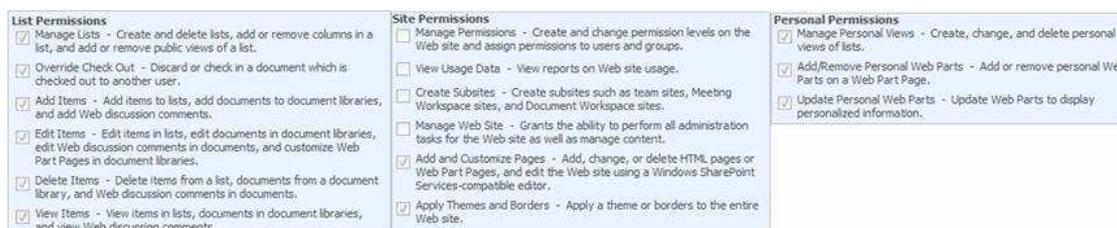


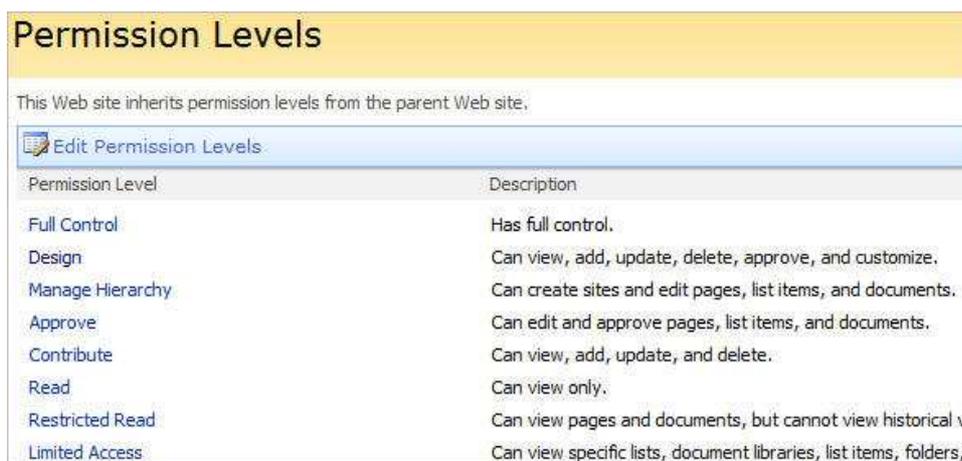
Figura 32 – Lista de permissões no Sharepoint

Níveis de permissão são um grupo de permissões que podem ser dadas a utilizadores ou grupos, de modo a que consigam efectuar certos tipos de tarefas em alguns tipos de objectos, tal como listas, bibliotecas, pastas ou documentos. Níveis de permissões permitem que possam ser agrupadas permissões e aplicá-las a utilizadores ou grupos.

Por defeito temos:

- Full control: Permite o acesso completo e controlo total sobre um site. É o nível menos restrito.
- Leitura – permite apenas visualizar

- Design: permite visualizar, adicionar, alterar, remover, aprovar e customizar listas, bibliotecas e páginas.
- Contribute: permite visualizar, adicionar, alterar e remover listas e documentos.
- Limited Access: é atribuída por defeito a utilizadores e grupos. É utilizada para permitir acesso a objectos child em cujo parent não tem acesso.



Permission Level	Description
Full Control	Has full control.
Design	Can view, add, update, delete, approve, and customize.
Manage Hierarchy	Can create sites and edit pages, list items, and documents.
Approve	Can edit and approve pages, list items, and documents.
Contribute	Can view, add, update, and delete.
Read	Can view only.
Restricted Read	Can view pages and documents, but cannot view historical versions.
Limited Access	Can view specific lists, document libraries, list items, folders, and documents.

**Figura 33 – Níveis de permissões**

Securable Objects – O Sharepoint permite que sejam dadas permissões a objectos individuais como sites, listas ou bibliotecas, até mesmo ao nível do item da lista, pastas, etc., independentemente das permissões definidas no parent ou nível superior.

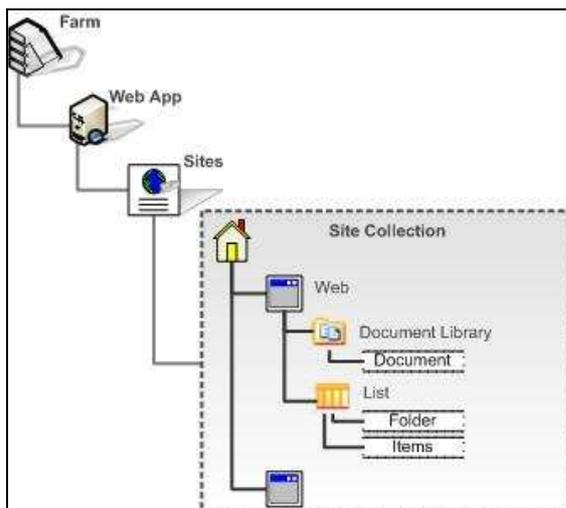


Figura 34 – Securable Objects

### A.2.10 Hierarquia e Herança

No Sharepoint, as permissões em qualquer objecto, sejam sites, listas, bibliotecas ou pastas, são herdadas do objecto parent. Contudo, podemos quebrar essa herança a qualquer objecto a um nível inferior da hierarquia. Podemos por exemplo, criar um Subsite e quebrar essa herança se quisermos definir as permissões manualmente. Todos os objectos nesse nível ou abaixo desse Subsite terão as permissões manuais. Estas características não se reflectem no sentido inverso. Conforme imagem abaixo, os Subsites A,B e C herdam do parent Top Level. Qualquer alteração no parent vai se reflectir nos childs, mas o D possui permissões distintas que não são herdadas.



Figura 35 – Relações de herança

### A.2.11 Utilizadores e grupos de Sharepoint

Tendo um utilizador autenticado, podemos atribuir permissões ao mesmo, mas será sempre muito mais fácil fazer esta gestão sobre um grupo a que este utilizador pertença, pois alterando um grupo, todos os utilizadores reflectirão essas alterações. Todos os grupos são criados ao nível da site collection<sup>9</sup> e estão disponíveis a todos os seus subsites. Estes grupos podem conter grupos de segurança Windows (AD), ASP.NET FBA através de Roles) ou a utilizadores individuais com uma conta criada no servidor local ou num domínio.

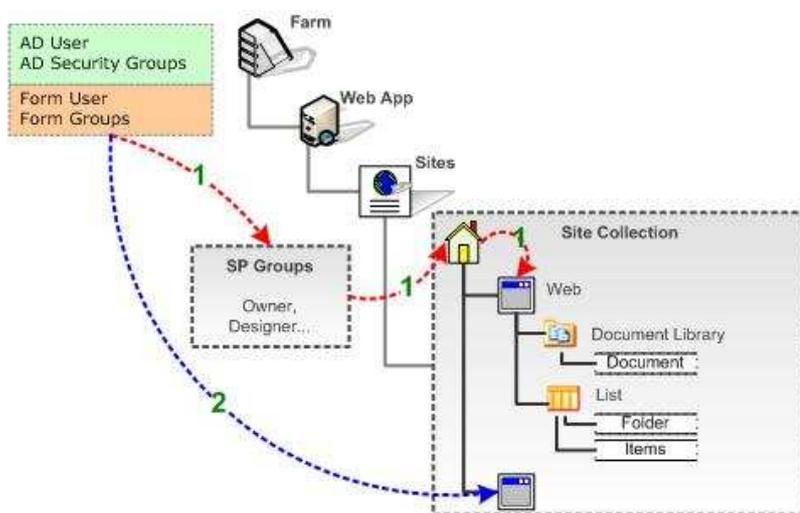


Figura 36 – Utilizadores e Grupos no Sharepoint

### A.2.12 Tipos de conteúdos e funcionalidades

Os Tipos de conteúdo permitem às organizações organizar, gerir e tratar conteúdos de modo consistente numa colecção de sites. Definindo tipos de conteúdo para tipos específicos de documentos ou produtos de informação, uma organização pode garantir que cada um desses grupos de conteúdo é gerido de modo consistente. É um grupo reutilizável de definições para uma categoria de conteúdo permitindo metadados, modelos e comportamentos de itens e documentos consistentemente. Os tipos de conteúdo são definidos ao nível do site e são utilizados em listas e bibliotecas. Cada tipo de conteúdo pode especificar:

<sup>9</sup>Colecção de sites: conjunto de Web sites num servidor virtual que têm o mesmo proprietário e que partilham definições de administração. Cada colecção de sites contém um Web site de nível superior e pode conter um ou vários subsites.

- As colunas (metadados) que pretende atribuir a itens deste tipo.
- O modelo de documento no qual pretende basear novos itens deste tipo (apenas tipos de conteúdo de documento).
- Os formulários Novo, Editar e Apresentar a utilizar com este tipo de conteúdo.
- Os fluxos de trabalho disponíveis para itens deste tipo de conteúdo.
- As funcionalidades ou soluções personalizadas associadas a itens deste tipo de conteúdo.
- As políticas de gestão de informações associadas a itens deste tipo de conteúdo.

Os tipos de conteúdo são organizados numa hierarquia que permite que um tipo de conteúdo herde as respectivas características de outro tipo de conteúdo. Tal como na hierarquia de sites, podemos alterar estas características de um modo particular se necessário.

### **A.2.13 Site Content Type**

O Site Content Type é um template do Sharepoint que contém informação com o template do documento, a metadata de promoted properties e workflows. Deste modo pode ser replicado por várias bibliotecas de documentos, facilitando o deployment das soluções, pois ao se alterar o template, as alterações vão se repercutir por todas as bibliotecas com base nesse template .

### **A.2.14 Bibliotecas e Listas de Sharepoint**

Uma Lista de Sharepoint é uma colecção de itens semelhantes, contendo colunas ou campos que definem os dados do item. São definidos por metadata e podemos equipará-las a uma tabela de uma base de dados na sua estrutura e comportamento. Podemos filtrar, ordenar, agrupar e efectuar pesquisas em listas através de CALM (uma query language baseada em XML).

Uma biblioteca é uma localização onde se pode criar, recolher, actualizar e gerir ficheiros ou documentos. Cada biblioteca apresenta uma lista de ficheiros e informações-chave sobre estes. Pode-se criar e gerir documentos, imagens, folhas de cálculo, apresentações, formulários e outros tipos de ficheiros. Uma biblioteca de documentos permite conter todo o tipo de documentos podendo no entanto definir-se um template<sup>10</sup> de modo a, ao criar um novo documento via Sharepoint, o programa usado pelo template abrir-se-á, podendo-se posteriormente guardar o ficheiro para esta mesma biblioteca. Tanto as listas como as bibliotecas, podem ter controlo de versões, onde podemos armazenar, controlar e restaurar itens conforme são alterados.

A biblioteca de Sharepoint é uma lista Sharepoint mas com características especiais. Numa lista normal, os metadados são o que define o item da lista, enquanto que numa biblioteca de documentos, o documento em si é o que define a lista.

Outra característica interessante de uma biblioteca de documentos é a property promotion, que permite que sejam visualizados dados fora do formulário, em colunas na biblioteca de formulários, como itens de uma lista.

No caso específico deste trabalho, será criada uma biblioteca de formulários, para cada inquérito, que inclui formulários XML criados em InfoPath como tipo de documento primário. Deste modo, o conceito e as features são partilhadas seja com InfoPath ou através do browser.

#### **A.2.15 Web Parts**

Ao incluir Web Parts e páginas Web Part o WSS disponibiliza uma infra-estrutura de criação de interfaces de utilizador (UI). Uma Web Part é um controlo ASP.NET que tem como objectivo expor informação. Web Parts podem ser adicionadas, movidas ou removidas, directamente no browser e preenchidas por outras Web Parts. Estas Web Parts podem ser inalteráveis ou controladas, adicionadas e removidas pelos próprios utilizadores. Deste modo, uma página pode ficar totalmente ao gosto do utilizador ou do grupo de trabalho.

---

<sup>10</sup> Template - Modelo de ficheiro que define um padrão. Todos os ficheiros criados pelo mesmo modelo, partilham características idênticas.

### **A.2.16 Workflows**

Um Workflow é uma forma de organizar e executar séries de acções que correspondem a um processo de trabalho que pode ter variados níveis de complexidade, dependendo das necessidades do processo de negócio.

O workflow é iniciado com base em eventos, que podem ser:

- Criação de um item numa lista
- Alteração de um Item numa lista

A acção é a base do workflow, é o que define o tipo de tarefa que irá realizar e podem ser tão variados como criar, alterar itens de uma lista, enviar um e-mail, criar tasks ou fazer cálculos.

As acções são sempre realizadas sob condições, onde está definida a lógica e as regras sob a qual irá ser realizada a acção.

## Anexo B. Web Service

Um Web service é uma API, que pode ser acedida pela rede e permite que sejam realizadas operações definidas, como retornar, criar, alterar ou remover dados numa base de dados. Permite que diferentes aplicações de diferentes fontes possam comunicar entre eles sem ser necessário alteração de código, visto que a base da comunicação é realizada em XML. Web Service é um módulo de software auto-descritivo, acessível através de protocolos como XML e SOAP, e usado remotamente por outras aplicações. É independente de plataformas, linguagens de programação e sistemas operativos. A figura abaixo demonstra um esquema para comunicação entre aplicações usando Web Services.

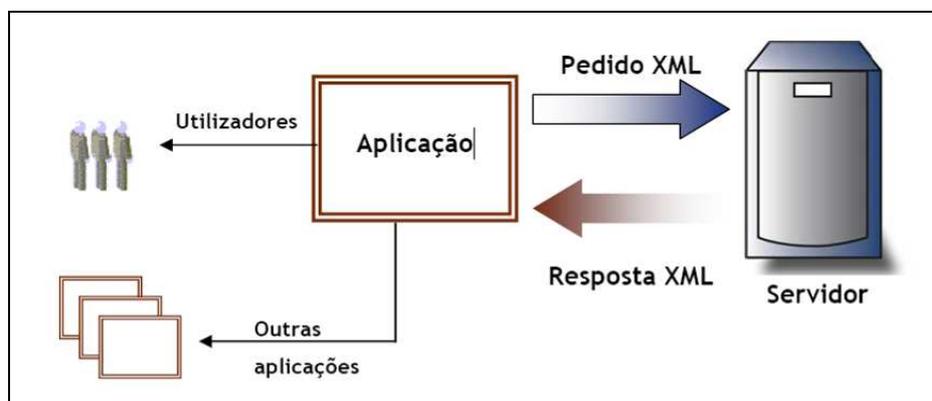


Figura 37 – Comunicação Web Services

As mensagens são codificadas utilizando XML, estruturadas utilizando SOAP e acedidas por HTTP, comunicando por meio de mensagens. É possível assim expor funcionalidades de programação na Web. A funcionalidade Web Service integrada com InfoPath, pode permitir o envio ou query de dados no mesmo formulário, de modo totalmente transparente para o utilizador.



Figura 38 - Comunicação InfoPath Web Service

## B.1 SOAP

O protocolo Simple Object Access Protocol é responsável pela estrutura das mensagens de um Web Service, tendo o objectivo de trocar informação em ambientes descentralizados e distribuídos. É baseado em XML/HTTP o que torna possível a disponibilização de mensagens pela Internet, sem estas estarem presas a qualquer linguagem ou sistema operativo. Numa mensagem SOAP, o Header contém a descrição da mensagem a ser enviada, enquanto que o Body contém o corpo da mensagem, formando-se assim o Envelope, que é a mensagem enviada em si. O Transport Envelope define como a mensagem será enviada.

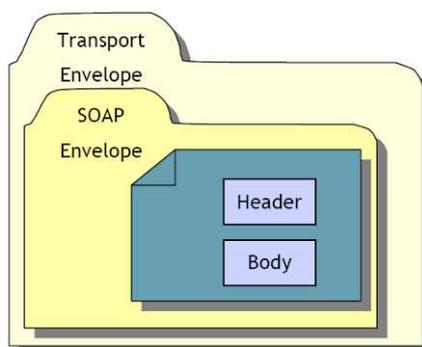


Figura 39 - Estrutura SOAP

## **Anexo C. Form Server e Microsoft Office InfoPath 2007**

Com a rápida e cada vez maior adopção de tecnologias baseadas em XML nos últimos anos, houve a necessidade de ferramentas que permitissem os utilizadores interagirem e partilharem dados em XML.

Com o InfoPath é possível criar e implementar soluções de formulários electrónicos para recolher informações de forma eficaz e fiável. Em conjunto com o Microsoft Office SharePoint Server 2007 podemos expandir os processos da organização para além das fronteiras da rede, entregando formulários como mensagens de correio electrónico do Microsoft Outlook, formulários de browser ou formulários para dispositivos móveis.

Com o InfoPath podem ser criados formulários dinâmicos, com diversas opções de visualização, de modo a recolher dados de qualquer tipo de audiência, sejam colaboradores, fornecedores, parceiros ou clientes que podem ser transformados em vários tipos de dados e utilizados por diferentes aplicações.

Podem ser criados para um ambiente de trabalho corporativo com fins específicos, podendo estruturar modelos de formulários ligados a bases de dados ou integrados em sistemas empresariais existentes. Por exemplo, com o mesmo modelo de formulário, pode fornecer dados a um sistema CRM, um sistema de gestão de pequenas despesas e um sistema de agendamento.

### **C.1 Microsoft Office InfoPath 2007**

O InfoPath é baseado em Extensible Markup Language (XML<sup>11</sup>). Quando criamos um modelo de formulário, o InfoPath cria um ficheiro .xsn, que é um ficheiro CAB (.cab) que contém os ficheiros necessários ao funcionamento do formulário, tais como os ficheiros XML Schema<sup>12</sup> (XSD) e XSL Transformation<sup>13</sup> (XSLT).

---

<sup>11</sup> XML – Extensible Markup Language para documentos com informação estruturada, que permite várias finalidades, tal como navegação, criação e análise de conteúdos, etc. A grande vantagem desta linguagem é a sua facilidade de criar e de integrar com outros sistemas de forma flexível.

<sup>12</sup> XSD – XML Schema – Descrição do tipo de documento XML que permite que sejam definidas regras a que o documento XML deve obedecer para ser considerado válido de acordo com o schema.

O principal objectivo de utilizar o InfoPath é obter dados de uma forma estruturada. Esta estrutura é representada por um schema (XSD) que define os campos de dados e a hierarquia da estrutura dos dados. Quando um utilizador preenche um formulário no InfoPath, os dados desse formulário são guardados ou submetidos por Data Connections definidas no template como XML standard no formulário e utilizados em *runtime*.

Sendo uma linguagem XML, pode ser reconhecida por múltiplos mecanismos como uma Web application, um Webservice ou integrar com diferentes aplicações como Excel, Word ou Access. Deste modo, providencia um formato de dados semi-estruturado, legível, processável por outras aplicações e funciona através de diferentes sistemas.

Utilizando o XML como linguagem de construção dos formulários, possibilita a criação de botões, *drop-down-menus*, caixas de texto, secções repetitivas, tabelas repetitivas, check boxes, *radio buttons*, entre outros tipos de controlos.

As grandes vantagens de utilizar o InfoPath são:

- **Maiores possibilidades de acesso** – É possível criar modelos de formulários compatíveis com o browser no InfoPath e activá-los de modo a poderem ser utilizados em Web sites internos e externos. Assim é possível partilhar formulários a diferentes tipos de audiências onde os utilizadores não têm de ter o InfoPath instalado no computador para preencher um formulário, nem é necessário que transfiram outro programa adicional da Web, bastando ter um browser e inclusivamente permite que os utilizadores o possam abrir e preencher utilizando um dispositivo móvel. Se os utilizadores tiverem o InfoPath instalado nos computadores, podem visualizar e preencher o formulário no InfoPath em vez de o fazerem num browser.
- **Integração com aplicações Office** - O InfoPath integra-se com outras aplicações como o Microsoft Outlook, Microsoft Excel e Microsoft Access. Deste modo podemos, por exemplo, estruturar e preencher formulários do InfoPath em mensagens de correio electrónico, exportar dados do formulário para folhas de

---

<sup>13</sup> XSLT – XLS Transformation – é o componente XSL que permite que se criem stylesheets aplicáveis a documentos XML de modo a transformar documentos XML noutros tipos como HTML.

cálculo, submeter dados do formulário para uma base de dados ou consultar dados a partir de uma lista do SharePoint.

- **Dados reutilizáveis** Os dados introduzidos num formulário InfoPath não ficam restritos a esse formulário; podem ser reformatados ou reutilizados de várias formas inclusive em outros processos já existentes. Isto permite que a informação seja sempre actualizada e disponível em toda a organização, mesmo que seja em sistemas pré-existentes.
- **Validação e consistência da informação** - O InfoPath inclui funcionalidades de controlo de introdução de dados que evitam erros no preenchimento, desde verificação ortográfica, a campos calculados, e com formatação condicional que informa o utilizador do erro cometido. Um formulário com erros nunca é submetido, assegurando que os dados recolhidos são exactos e que respeitam os padrões especificados no desenho em InfoPath. Permite formatação condicional, podendo ser definidos padrões de texto a serem introduzidos e ainda ser definidas regras que caso certas condições aconteçam, como por exemplo um determinado campo conter um determinado valor, acções específicas podem ser realizadas tal como definir um valor num campo, fazer uma query, ou submeter ou obter valores a uma Data Connection, etc.
- **Suporte off-line** – Os formulários do InfoPath não têm de ser preenchidos com ligação a uma rede. Os utilizadores podem guardar formulários localmente, trabalhar neles off-line e, em seguida, submeter um ou mais formulários para a plataforma da organização. Esta funcionalidade é particularmente útil para o caso de acessos Internet e utilizadores remotos.
- **Vistas de formulários** – É possível através do mesmo modelo de formulário criar várias vistas, dependendo do perfil do utilizador, o que facilita a sua administração. Por exemplo, num mesmo modelo de formulário podemos ter uma vista para os funcionários e uma segunda vista para os gestores que aprovam. Deste modo, com o mesmo registo, a informação fica centralizada, sem duplicação e sem fugas de informação.

- **Data Connections** – Utilizando o InfoPath, os dados pode ser acessíveis através de data connections, que são definidas no próprio formulário e utilizadas em runtime. A data connection é uma ligação dinâmica entre o InfoPath e uma fonte de dados externa que armazena ou disponibiliza os dados para o formulário. Um formulário pode ter uma data connection principal e outras secundárias que podem aceder a vários tipos de serviços e realizar diferentes tarefas, dependendo da estrutura implementada do formulário. Pode inclusive no mesmo formulário existir os dois tipos de data connection, onde os dados são recebidos por uma query e submetidos depois do preenchimento. Na criação do formulário, o administrador define a data connection a utilizar e podemos dividi-la em dois tipos:
- **Submit data connection** – A data connection é associada a uma acção do utilizador, como por exemplo um botão em que os dados do formulário são enviados para o alvo da data connection. Esses dados podem ser enviados para:
  - Web Service – Os dados são enviados Segundo as especificações do web service
  - Biblioteca de documentos SharePoint – Os dados são enviados para uma biblioteca de documentos utilizando uma data connection ligada ao SharePoint Document Library adapter.
  - E-mail – Através de uma data connection ligada ao SMTP E-Mail adapter podemos enviar para um endereço de mail.
  - Página ASP.NET ou uma aplicação – Com uma ligação ao adapter “Submeter para host” podemos accionar um evento para o host e esse host é responsável pelo tratamento dos dados do formulário.
  - Web Server (HTTP) – A definição do envio especifica um endereço http e o Post é usado para enviar os dados do formulário para o url.
  - Base de Dados – Se o formulário for desenhado para uma base de dados a acção de submeter pode enviar directamente os dados para a mesma.

As opções disponíveis para submeter dados para um formulário Web são apenas as de Web service, Biblioteca de documentos SharePoint, E-mail, Web Server (HTTP). A opção base

de dados não é possível num formulário Web por limitações da plataforma devido a questões de segurança.

- **Receive (query) data connection** – Esta data connection permite que ao abrir o formulário, seja feita uma query, e onde os dados retornados podem ser obtidos directamente de:
  - Documento XML – qualquer documento que retorne XML, por exemplo um feed RSS.
  - Base de Dados
  - Web Service
  - Biblioteca de documentos SharePoint

Através da data connection de query, podemos disponibilizar no formulário dados sempre actualizados e garantir consistência no envio. Um exemplo prático é na abertura do formulário a query preencher os dados de uma dropdown list, que pode ser actualizada conforme os dados recebidos anteriormente sem que seja necessário defini-los á mão.

## **C.2 Integração InfoPath e SQL Server**

A integração directa de um formulário Web de InfoPath e uma base de dados SQL não é possível devido a questões de segurança. Quando o utilizador abre o formulário no browser (1º nível) o formulário comunica com o Forms Services (2º nível) que por sua vez acede à base de dados (3º nível). O problema existe no token<sup>14</sup> de segurança que é utilizado entre o 1º nível e o 2º nível não conseguem chegar até ao servidor de base de dados devido a dois problemas:

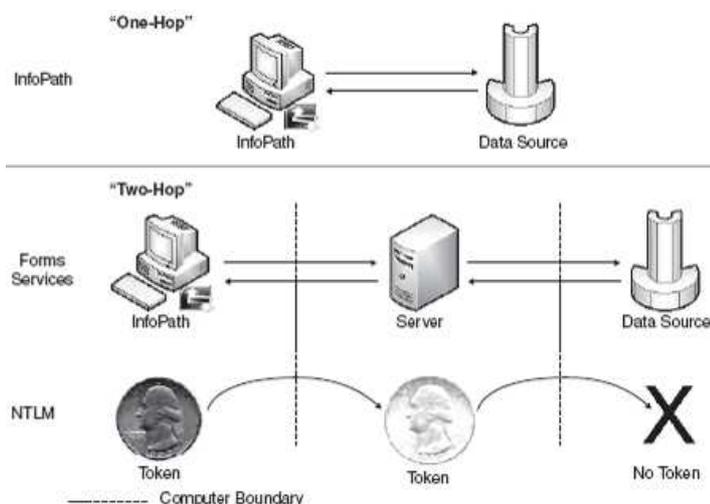
- Restrições relativas ao sistema de credenciais de autenticação que necessita de aceder a recursos num 3º nível a partir de recursos de um 1º ou 2º nível.

---

<sup>14</sup> Token – Chave de prova de identidade electrónica de entrada num sistema, utilizada como modo de segurança para facilitar a entrada no sistema a um utilizador autenticado com o máximo de segurança.

- Problema com cruzamento de Domínios<sup>15</sup>. Se o InfoPath estiver a enviar ou receber dados de um web service em <http://inquerito.uatla.pt/service.aspx> e a Plataforma Sharepoint ou o Servidor da base de dados estiver num <http://abc.com> ou noutra servidor, o InfoPath está a tentar comunicar entre plataformas onde não existe ligação de *Trust* com o servidor de Forms Services, não sendo assim possível enviar ou receber dados.

Existem resoluções para estes problemas já disponibilizadas pelo fabricante, tal como SSO<sup>16</sup> (*Single Sign On*) e o UDC<sup>17</sup> (*Universal Data Connections*) para gerir a autenticação entre arquitecturas multi-nível e utilizando o Web Service Proxy disponível com o Forms Services.



**Figura 40 - Restrições InfoPath**

Apesar de tudo, algumas destas opções podem ser consideradas problemáticas na implementação tanto a nível de design do template, assim como da administração da plataforma e finalmente alguns problemas de segurança.

<sup>15</sup> Cross Domain Issue – Problema de autenticação com a passagem entre diferentes Domínios ou sistemas de autenticação.

<sup>16</sup> SSO – Serviço que armazena e mapeia credenciais para utilização com aplicações/servidores num 3º nível.

<sup>17</sup> UDC – Biblioteca de ligação de dados centralizada que pode conter ligações do tipo Office Data e UDC que permitem que um cliente ou um formulário web possa aceder a diferentes Data Sources.

Não é objectivo deste trabalho a descrição destas hipóteses de resolução pelo que, devido á arquitectura seguida e características da plataforma instalada, não foi adoptada nenhuma destas soluções.

### **C.3 Forms Services**

O Forms Services<sup>18</sup> é a feature do Sharepoint Server 2007 que permite a gestão e administração centralizada de formulários. Está inclusive dependente da plataforma para realizar tarefas tal como o deployment, controlo de versões, etc.

Com o Forms Services, pretende-se dar o mesmo tipo de funcionalidades e serviços a um utilizador Web ou a um utilizador com InfoPath, onde o mesmo modelo de formulário pode ser utilizado sem qualquer tipo de diferenças para o utilizador através de um browser. Apesar de a versão Web ser um pouco mais simplificada, em nada limita ou impede a utilização desta opção que no interface para o utilizador pode ser praticamente idêntico.

Com o Forms Services temos a possibilidade de aceder e preencher um formulário num browser, tendo ou não o cliente InfoPath instalado, num sistema operativo Windows ou não, ou até mesmo com um computador ou um *smart phone*, sempre com o mesmo modelo de formulário. Com a base na plataforma Sharepoint Server 2007, o Forms Services pode integrar com bibliotecas de documentos, workflows, segurança e administração.

O Form Services, integrado com o MOSS permite que independentemente do sistema do utilizador, o serviço seja disponibilizado e gerido de forma centralizada, facilitando a gestão e qualidade do serviço.

#### **C.3.1 Opção Form Services para Web browser**

Apesar da opção de utilizar o cliente InfoPath ser mais a interessante, é bastante mais limitativa, pois existem casos que impedem a sua utilização, tal como:

---

<sup>18</sup> Forms Server é um nome de produto standalone (marketing), Forms Service é um componente Server incluído no Sharepoint Server 2007 com Enterprise Features activo.

- Mesmo num ambiente interno da organização pode ser impossível instalar o cliente InfoPath em todas as máquinas, o que se torna evidente se, tal como no caso deste trabalho, se pretende dar visibilidade de um formulário através da Internet a utilizadores que não podemos saber se possuem ou não o cliente. Não é nosso objectivo forçar a compra e instalação de licenças de software, muito pelo contrário.
- Não é possível esperar, e muito menos forçar, que todos os intervenientes da organização, sejam eles clientes, colaboradores ou qualquer outra pessoa externa á organização que pretenda ter acesso ao formulário, tenha o cliente InfoPath instalado.
- Sistemas operativos como o Linux não têm qualquer versão InfoPath disponível.

Utilizando o Forms Services na plataforma Sharepoint, com o acesso através de um Web browser este problema deixa de existir:

- Máquinas com recursos mínimos abaixo do exigido para instalação do InfoPath, basta que tenham um browser para funcionar.
- Máquinas que não possuem sequer um sistema operativo Windows, seja Apple ou UNIX, e cuja instalação do InfoPath seria impossível, podem funcionar através de um browser.
- Equipamentos como *smart phones* podem aceder e funcionar através de um browser.

Em suma, qualquer equipamento, desde que possua um browser e acesso á rede ou Internet pode, do lado do cliente, aceder ao um formulário, e ter praticamente o mesmo nível de serviço e capacidades. Do lado da gestão de dados e da informação recolhida, seja utilizando o InfoPath ou o browser, os dados são idênticos.

### **C.3.2 Requisitos Mínimos para o Form Services**

O Forms Service, visto disponibilizar um serviço orientado para um acesso Web, está concebido para correr apenas em Web Front Ends.

- SharePoint Server requer o Windows Server 2003 SP1 ou superior
- Internet Information Services (IIS)
- .NET Framework 2.0 activada no IIS
- Windows Workflow Foundation

### C.3.3 Browsers Suportados

Sem dúvida que aqui poderia haver alguma discussão sobre qual o melhor browser, mas principais browsers são suportados, conforme a lista abaixo:

Plataforma	Browser	Versão	Observações
Windows	Internet Explorer	6 ou superior	Sem restrições
Windows	Firefox	1.5	Assinaturas digitais com apenas com IE
Windows	Netscape	8.1	Rich text box é read only mas configurável no browser
Windows	Mozilla	1.7	
Mac Os	Firefox	1.5	Sem opção de auto-complete.
Mac OS	Safari	2.0	Botão de funcionalidade de escolha de data inacessível
UNIX/Linux	Firefox	1.5	
UNIX/Linux	Netscape	7.2	

**Tabela 20 – Tabela de browsers compatíveis**

Apesar de suportar todos os diferentes browsers, o fabricante recomenda a utilização do Internet Explorer 6 ou superior.

## Anexo D. Plataforma a instalar

Respeitando os requisitos mínimos para instalação da plataforma, foi definido que devido a restrições no ambiente de produção, não seria possível a existência de uma arquitectura optimizada com uma Farm para a plataforma devido aos impacto que poderia provocar na estrutura existente.

Optou-se por uma configuração single-server, que é a mais restrita possível. Deste modo, foi instalado numa máquina toda a plataforma Sharepoint e SQL Server. Considerando que é um ambiente bastante reduzido e com uma carga bastante baixa a nível de utilizadores e quantidade de dados, apesar de tudo pensamos ser suficiente para os objectivos a atingir. As características de uma instalação deste tipo são as seguintes:

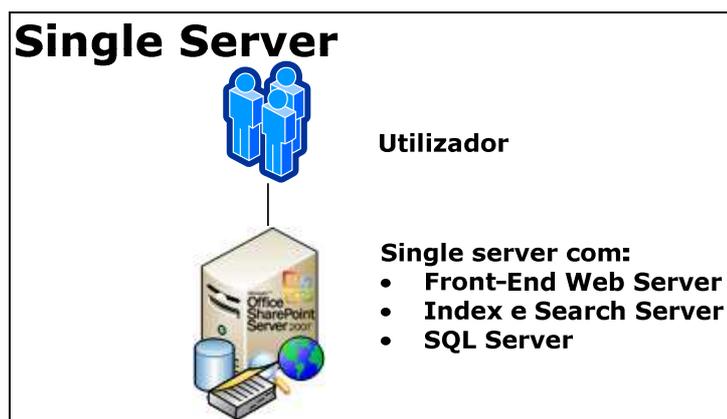


Figura 41 – Arquitectura a instalar

- Todos os Server roles são realizados por um único Server, por esta razão, a performance será limitada.
- A disponibilidade será limitada, pois a fault-tolerance e load balancing não são implementados.
- Podem ser criados vários sites.
- É adequado a workgroups, equipas e testes, mas não se recomenda numa estrutura Organizacional.

- Sharepoint Sever com Enterprise features, de modo a disponibilizar o Forms Services.
- Acesso pela rede interna e rede Internet sob configuração no ISA Server da organização.
- No mesmo servidor existem a plataforma instalada, o Web front-end Server, a Application Server e a base de dados.

Foram instalados numa Virtual Machine o Sistema Operativo Windows Server 2003 SP2, com SQL Server 2008, Microsoft Office Sharepoint Server 2007 com SP1 e Enterprise Features, Language Pack SP2 português, Visual Studio 2008, e Microsoft Office 2007.

Devido aos recursos existentes na organização, irão existir algumas restrições a que este projecto necessitou de se adaptar, e onde sempre que possível, irá ser referido o porquê da opção tomada e quais as opções disponíveis. O sistema operativo base foi instalado numa Virtual Machine inserida no domínio da Organização e os requisitos mínimos indicados pelo fabricante foram respeitados. As tecnologias utilizadas no desenvolvimento da solução são as seguintes:

- Microsoft Windows Server 2003 Standard Edition com SP2 32 bit
- Internet Information Services 6.0
- ASP.NET 2.0 e 3.5
- Microsoft Visual Studio 2008 Professional Edition
- Microsoft SQL Server 2008
- Microsoft Office Sharepoint Server 2007
- Microsoft Office Server Language Package 2007 PT
- Microsoft Office 2007
- Office InfoPath 2007
- Microsoft Office Access 2007
- Microsoft Office Excel 2007
- Microsoft Office Word 2007
- Web Services desenvolvidos no Microsoft Visual Studio 2008 C#

## Anexo E. Código e Scripts Utilizados

Aqui se apresentam os scripts SQL relativos á criação da base de dados e da Tabela Inquéritos, assim como o código-fonte em linguagem C# do web service criado.

### E.1 Script SQL

Aqui se apresentam os scripts SQL relativos á criação da Base de Dados “Dados” e da tabela Inquéritos.

#### E.1.1 Script SQL da Base de Dados

No quadro seguinte, encontra-se o script de criação da Base de Dados de nome “Dados” utilizada pelo Web Service:

```
USE [master]
GO
/***** Object: Database [Dados] Script Date: 11/10/2009 20:02:15
*****/
CREATE DATABASE [Dados] ON PRIMARY
( NAME = N'dados', FILENAME = N'C:\Program Files\Microsoft SQL
Server\MSSQL10.MSSQLSERVER\MSSQL\DATA\dados.mdf' , SIZE = 2048KB ,
MAXSIZE = UNLIMITED, FILEGROWTH = 1024KB )
LOG ON
( NAME = N'dados_log', FILENAME = N'C:\Program Files\Microsoft SQL
Server\MSSQL10.MSSQLSERVER\MSSQL\DATA\dados_log.ldf' , SIZE = 1024KB ,
MAXSIZE = 2048GB , FILEGROWTH = 10%)
GO

ALTER DATABASE [Dados] SET COMPATIBILITY_LEVEL = 100
GO

IF (1 = FULLTEXTSERVICEPROPERTY('IsFullTextInstalled'))
begin
EXEC [Dados].[dbo].[sp_fulltext_database] @action = 'enable'
end
GO
ALTER DATABASE [Dados] SET ANSI_NULL_DEFAULT OFF
GO
ALTER DATABASE [Dados] SET ANSI_NULLS OFF
GO
ALTER DATABASE [Dados] SET ANSI_PADDING OFF
GO
ALTER DATABASE [Dados] SET ANSI_WARNINGS OFF
GO
ALTER DATABASE [Dados] SET ARITHABORT OFF
```

```
GO
ALTER DATABASE [Dados] SET AUTO_CLOSE OFF
GO
ALTER DATABASE [Dados] SET AUTO_CREATE_STATISTICS ON
GO
ALTER DATABASE [Dados] SET AUTO_SHRINK OFF
GO
ALTER DATABASE [Dados] SET AUTO_UPDATE_STATISTICS ON
GO
ALTER DATABASE [Dados] SET CURSOR_CLOSE_ON_COMMIT OFF
GO
ALTER DATABASE [Dados] SET CURSOR_DEFAULT GLOBAL
GO
ALTER DATABASE [Dados] SET CONCAT_NULL_YIELDS_NULL OFF
GO
ALTER DATABASE [Dados] SET NUMERIC_ROUNDABORT OFF
GO
ALTER DATABASE [Dados] SET QUOTED_IDENTIFIER OFF
GO
ALTER DATABASE [Dados] SET RECURSIVE_TRIGGERS OFF
GO
ALTER DATABASE [Dados] SET DISABLE_BROKER
GO
ALTER DATABASE [Dados] SET AUTO_UPDATE_STATISTICS_ASYNC OFF
GO
ALTER DATABASE [Dados] SET DATE_CORRELATION_OPTIMIZATION OFF
GO
ALTER DATABASE [Dados] SET TRUSTWORTHY OFF
GO
ALTER DATABASE [Dados] SET ALLOW_SNAPSHOT_ISOLATION OFF
GO
ALTER DATABASE [Dados] SET PARAMETERIZATION SIMPLE
GO
ALTER DATABASE [Dados] SET READ_COMMITTED_SNAPSHOT OFF
GO
ALTER DATABASE [Dados] SET HONOR_BROKER_PRIORITY OFF
GO
ALTER DATABASE [Dados] SET READ_WRITE
GO
ALTER DATABASE [Dados] SET RECOVERY FULL
GO
ALTER DATABASE [Dados] SET MULTI_USER
GO
ALTER DATABASE [Dados] SET PAGE_VERIFY CHECKSUM
GO
ALTER DATABASE [Dados] SET DB_CHAINING OFF
GO
```

**Figura 42 - Script de criação da Base de Dados**

### E.1.2 Script SQL da Tabela Inquerito

No quadro seguinte, encontra-se o script de criação da tabela Inquerito da Base de Dados utilizada pelo Web Service:

```
USE [Dados]
GO
/***** Object: Table [dbo].[Inquerito]    Script Date: 11/10/2009
19:59:33 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
SET ANSI_PADDING ON
GO
CREATE TABLE [dbo].[Inquerito](
    [Id_Form] [int] NOT NULL,
    [Cod_Pais] [varchar](50) NULL,
    [Cod_Escola] [int] NULL,
    [Nome_Escola] [varchar](50) NULL,
    [Rua_Escola] [varchar](50) NULL,
    [Cidade_Escola] [varchar](50) NULL,
    [Regiao_Escola] [varchar](50) NULL
) ON [PRIMARY]
SET ANSI_PADDING OFF
GO
```

**Figura 43 - Script de criação da Tabela**

Esta tabela não é representativa de toda a Base de dados necessária para um formulário de Inquérito, pois visto esta solução não ter sido implementada, esta tabela foi criada apenas em parte de modo a ser possível demonstrar a possibilidade de solução.

### E.1.3 Código-fonte C# do Web Service

No quadro seguinte, encontra-se a estrutura e código desenvolvido em C# para a criação do Webservice de ligação á base de dados. Neste Web Service existem cinco métodos para efectuar as diferentes operações necessárias descritas na tabela abaixo:

Método	Descrição
DeleteInqueritoByID	Método de Remoção de inquerito por ID de Formulário
UpdateInqueritoByID	Método de Update inquerito por ID de Formulário
GetAllInqueritos	Método de Select inquerito que retorna todos os Formulários
AddNewInquerito	Método que insere novo inquerito
GetInqueritoByID	Método de Select inquerito por ID de Formulário

**Tabela 21 - Métodos do Web Service**

O código desenvolvido na criação do Web Service foi desenvolvido em C# .NET Framework 3.5, em Microsoft Visual Studio 2008 e está descrito e comentado no seguinte quadro:

```
using System;
using System.Web;
using System.Web.Services;
using System.Web.Services.Protocols;
using System.Data.Sql;
using System.Data.SqlClient;
using System.Xml;
using System.Data;
using System.Configuration;

namespace Service
{
    /// <summary>
    /// Este Web service tem o objectivo de permitir através de InfoPath
    operações de inserção e update e delete numa base de
    ///dados instalada numa base de dados SQL no servidor com o ip:
    192.168.10.33, na instancia ceidss, bd:
    ///Dados, tabela Inqueritono ambito dos projectos de investigação da
    plataforma Sharepoint instalada para os
    /// projectos de investigação
    /// </summary>

    [WebService(Namespace = "http://tempuri.org/")]
    [WebServiceBinding(ConformsTo = WsiProfiles.BasicProfile1_1)]
```

```

public class Servicel : System.Web.Services.WebService
{
    private SqlConnection sqlConnection1;

    public struct Inq
    {
        public int idForm;
        public String codPais;
        public String codEscola;
        public String nomeEscola;
        public String ruaEscola;
        public String cidadeEscola;
        public String regioaoEscola;
    }

    //Método de select inquerito por ID de Formulário
    [WebMethod]
    public XmlDocument GetInqueritoByID(string FormID)
    {
        XmlDocument doc = new XmlDocument();
        DataSet InqDataSet = new DataSet("Inquerito");

        //String de ligação á base de dados
        using (SqlConnection conn = new
SqlConnection("server=CEIDSS;uid=sas;pwd=devteam;database=Dados"))
        {
            //Criação da string de operação SQL e ligação
            SqlDataAdapter adapter = new SqlDataAdapter("Select
Id_Form, Cod_Pais, Cod_Escola, Nome_Escola, Morada_Escola, Rua_Escola,
Cidade_Escola, Regiao_Escola from Inquerito where Id_Form = '" + FormID +
"'", conn);

            //preenchimento do Dataset
            adapter.Fill(InqDataSet, "Inquerito");
        }

        doc.LoadXml(InqDataSet.GetXml());

        //retorna o conteudo do Dataset em XML
        return doc;
    }

    //Método de Remoção de inquerito por ID de Formulário
    [WebMethod]
    public XmlDocument DeleteInqueritoByID(string FormID)
    {
        XmlDocument doc = new XmlDocument();
        DataSet InqDataSet = new DataSet("Inquerito");
        using (SqlConnection conn = new

        //String de ligação á base de dados
SqlConnection("server=CEIDSS;uid=sas;pwd=devteam;database=Dados"))

```

```

        { //Criação da string de operação SQL e ligação
          SqlDataAdapter adapter = new SqlDataAdapter("Delete from
Inquerito where ID_Form ='" + FormID + "'", conn);
          //preenchimento do Dataset
          adapter.Fill(InqDataSet, "Inquerito");
        }

        doc.LoadXml(InqDataSet.GetXml());

        return doc;
    }

    //Método de update inquerito por ID de Formulário
    [WebMethod]
    public XmlDocument UpdateInqueritoByID(int idForm, string
codPais, string codEscola, string nomeEscola, string ruaEscola, string
cidadeEscola, string regioaEscola, Inq formID)
    {

        XmlDocument doc = new XmlDocument();
        DataSet InqDataSet = new DataSet("Inquerito");
        //String de ligação á base de dados
        using (SqlConnection conn = new
SqlConnection("server=CEIDSS;uid=sas;pwd=devteam;database=Dados"))
        {
            //Criação da string de operação SQL e ligação
            SqlDataAdapter adapter = new SqlDataAdapter("Update
Inquerito set Id_Form ='" + idForm + "',Cod_Pais='" + codPais +
"',Cod_Escola='" + codEscola + "', Nome_Escola='" + nomeEscola + "',
Rua_Escola='" + ruaEscola + "', Cidade_Escola='" + cidadeEscola + "',
Regiao_Escola='" + regioaEscola + "' where Id_Form = '" + idForm + "'",
conn);

            //preenchimento do Dataset
            adapter.Fill(InqDataSet, "Inquerito");

        }

        doc.LoadXml(InqDataSet.GetXml());
        return doc;
    }

    //Método de Select inquerito que retorna todos os Formulários
    [WebMethod]
    public XmlDocument GetAllInqueritos()
    {
        XmlDocument doc = new XmlDocument();
        DataSet InqDataSet = new DataSet("Inquerito");

        //String de ligação á base de dados
        using (SqlConnection conn = new
SqlConnection("server=CEIDSS;uid=sas;pwd=devteam;database=Dados"))
        {
            //Criação da string de operação SQL e ligação
            SqlDataAdapter adapter = new SqlDataAdapter("Select
Id_Form, Cod_Pais, Cod_Escola, Nome_Escola, Morada_Escola, Rua_Escola,
Cidade_Escola, Regiao_Escola from Inquerito ", conn);
            //preenchimento do Dataset
            adapter.Fill(InqDataSet, "Inquerito");
        }
    }

```



## **Anexo F. Manual de utilizador**

Foi criado um manual de utilizador para ser disponibilizado aos membros do projecto e encontra-se disponível on-line na plataforma de produção no endereço: <http://ceidss.uatlantica.pt/munsi/ArtigosEstudos/Forms/AllItems.aspx>.

Neste manual estão descritos todos os passos para a utilização da plataforma como uma ferramenta de trabalho, incluindo algumas features que não fizeram parte deste trabalho, mas por terem sido consideradas ferramentas e opções interessantes foram adicionadas á plataforma.

## Anexo G. Visio da Estrutura Implementada

