

Universidade de Lisboa

Faculdade de Ciências

Departamento de Informática



POCAL – Regime Simplificado

Luís André Barreiros de Matos

Mestrado em Engenharia Informática

2007

Universidade de Lisboa

Faculdade de Ciências

Departamento de Informática



POCAL – Regime Simplificado

Luís André Barreiros de Matos

Projecto orientado pelo Prof. Dr. João Pedro Neto
e co-orientado por Emílio Martins

Mestrado em Engenharia Informática

2007

Declaração

Luís André Barreiros de Matos, aluno n.º 29068 da Faculdade de Ciências da Universidade de Lisboa, declara ceder os seus direitos de cópia sobre o seu relatório de projecto em Engenharia Informática, intitulado "POCAL – Regime Simplificado", realizado no ano lectivo de 2006/2007 à Faculdade de Ciências da Universidade de Lisboa para o efeito de arquivo e consulta nas suas bibliotecas e publicação do mesmo em formato electrónico na Internet.

FCUL, 19 de Novembro de 2007

Emílio Martins, supervisor do projecto de Luís André Barreiros de Matos, aluno da Faculdade de Ciências da Universidade de Lisboa, declara concordar com a divulgação do relatório do projecto em Engenharia Informática, intitulado "POCAL – Regime Simplificado".

Lisboa, 19 de Novembro de 2007

Resumo

Este relatório tem como objectivo, apresentar o trabalho realizado no âmbito da disciplina de Projecto, do 2º ano do Mestrado em Engenharia Informática pela Faculdade de Ciências da Universidade de Lisboa.

O projecto visou a implementação do POCAL regime simplificado. Esta aplicação foi desenvolvida de raiz e tem como objectivo complementar a oferta da empresa de produtos ao nível da Gestão Autárquica. Pretende-se, com esta solução, disponibilizar uma ferramenta que permita responder de uma forma simples, integrada e eficiente, às necessidades contabilísticas, de prestação de contas e de gestão administrativa das Juntas de Freguesia e outras entidades sujeitas ao regime simplificado do POCAL.

Este projecto foi desenvolvido por um equipa de quatro programadores e um gestor. Cada programador desenvolveu um módulo do POCAL. O meu trabalho passou por modelar e implementar o módulo de Gestão de Património, melhoramento e correcções no módulo de Recursos Humanos, administração e *BackOffice*. E ainda implementei um componente para importação de dados de cadastro para o módulo do SIADAP.

Todo o trabalho foi realizado nas instalações da empresa, em Lisboa, com uma carga horária de 40 horas semanais.

PALAVRAS-CHAVE:

POCAL, Tecnologia Microsoft, solução WEB, módulo de Cadastro, Autarquias

Conteúdo

Lista de Figuras	10
Lista de Tabelas	12
1. Introdução	14
1.1. Estrutura do Relatório	14
1.2. Apresentação do Projecto	15
1.3. Identificação e contextualização	15
1.4. Motivação	16
1.5. Objectivos	16
1.6. Instituição de Acolhimento	17
1.6.1. AmbiSIG – Ambiente e Sistemas de Informação Geográfica	17
1.7. Resumo	19
2. Planeamento do Projecto	20
2.1. Análise e justificação da viabilidade do projecto	20
2.2. Definição da equipa para a implementação do projecto	21
2.3. Definição e apresentação do plano e das estratégias para implementação do sistema	21
2.4. Metodologia e Calendarização	22
2.4.1. Metodologia	22
2.4.2. Calendarização	25
3. Arquitectura e ferramentas Utilizadas	31
3.1. Arquitectura	31
3.2. Ferramentas Utilizadas	33
3.2.1. Linguagens de Programação	33
3.2.2. Base de Dados	34
3.2.3. Outras Ferramentas	34
4. Trabalho Realizado	37
4.1. POCAL	37
4.2. Modelo de Dados	37
4.3. Trabalho Desenvolvido	38
4.4. Funcionalidades implementadas	42
5. Conclusão	50
6. Bibliografia	51
7. Referências WWW	52
8. Lista de Siglas e Acrónimos	53
9. Anexos	54

Lista de Figuras

<i>Fig. 1 - Modelo da qualidade adoptado na empresa.....</i>	<i>17</i>
<i>Fig. 2 - Organograma da empresa</i>	<i>18</i>
<i>Fig. 3 - Fases e Tarefas do processo de desenvolvimento de software</i>	<i>23</i>
<i>Fig. 4 - Arquitectura 3 camadas</i>	<i>31</i>
<i>Fig. 5 - Administração - permissões.....</i>	<i>40</i>
<i>Fig. 6 - BackOffice gestão de tabelas de metadados</i>	<i>41</i>
<i>Fig. 7 - Componente de importação</i>	<i>42</i>
<i>Fig. 8 - Interface inicial.....</i>	<i>43</i>
<i>Fig. 9 - Autenticação do utilizador.....</i>	<i>43</i>
<i>Fig. 10 - Contexto genérico</i>	<i>44</i>
<i>Fig. 11 - Módulo de Gestão de Património - feedback.....</i>	<i>45</i>
<i>Fig. 12 - Listagem dos bens móveis (CIME).....</i>	<i>45</i>
<i>Fig. 13 - Geração de etiquetas para identificação dos Bens.....</i>	<i>46</i>
<i>Fig. 14 - Ficha do Bem para consulta ou edição</i>	<i>46</i>
<i>Fig. 15 - Variações patrimoniais de um Bem</i>	<i>47</i>
<i>Fig. 16 - Listagens das amortizações</i>	<i>47</i>
<i>Fig. 17 - Ficha de identificação dos Bens móveis</i>	<i>48</i>
<i>Fig. 18 - Contexto mapas.....</i>	<i>48</i>

Lista de Tabelas

<i>Tabela 1 - Planeamento dos principais módulos do projecto</i>	<i>25</i>
<i>Tabela 2 - Aquisição dos conhecimentos dos processos adoptados pela empresa</i>	<i>26</i>
<i>Tabela 3 - Aquisição dos requisitos POCAL.....</i>	<i>26</i>
<i>Tabela 4 - Elaboração do modelo de dados</i>	<i>27</i>
<i>Tabela 5 - Implementação do Módulo de Cadastro</i>	<i>28</i>
<i>Tabela 6 - Verificação de conformidades.....</i>	<i>29</i>
<i>Tabela 7 - Melhoramentos e implementação de novas funcionalidades</i>	<i>30</i>

1. Introdução

Este relatório tem como objectivo apresentar o trabalho realizado no âmbito da disciplina de Projecto, do 2º ano do Mestrado em Engenharia Informática pela Faculdade de Ciências da Universidade de Lisboa.

Ao longo do período académico pretende-se que o aluno adquira um conjunto de competências, conhecimentos científicos e tecnológicos, que posteriormente serão postos em prática na vida profissional. A finalidade principal deste estágio foi aplicar os conhecimentos adquiridos ao longo do período académico, inserção na actividade profissional, e integração num ambiente de produção e aprofundamento dos conhecimentos técnicos / científicos.

Este relatório vai abordar diversos temas, desde a integração na empresa de acolhimento até à tecnologia utilizada, os processos de produção, a equipa de trabalho, e o meu contributo no projecto.

1.1. Estrutura do Relatório

O primeiro capítulo tem como objectivo apresentar a instituição de acolhimento e o projecto onde serão abordados vários aspectos desde as políticas adoptadas pela empresa, os seus processos, a sua estrutura organizacional, identificação e contextualização do projecto. O segundo capítulo descreve o planeamento do projecto e seus objectivos, a análise e a viabilidade do projecto, assim como a estrutura da equipa. O terceiro capítulo descreve a calendarização do projecto, nomeadamente as etapas estabelecidas e os tempos definidos. O quarto capítulo apresenta a arquitectura e ferramentas usadas. O quinto capítulo descreve o trabalho realizado na prática. No sexto capítulo são apresentadas as conclusões. No capítulo sétimo e seguintes são apresentadas a bibliografia, referências bibliográficas, lista de siglas e acrónimos, anexos e índice de figuras.

1.2. Apresentação do Projecto

O projecto proposto visa a criação de uma ferramenta que possibilite, de uma forma eficaz e intuitiva, a gestão de todos os processos inerentes ao POCAL - Plano Oficial de Contas das Autarquias Locais, num ambiente *web* e descentralizado. Este projecto foi uma aposta da empresa no desenvolvimento de um novo produto visto que não existia nenhum cliente associado inicialmente.

Todo o projecto foi implementado e desenvolvido nas instalações da empresa, com uma carga horária de 40 horas semanais.

O meu contributo neste projecto consistiu na modelação e implementação do módulo de Cadastro de Inventário de Bens de Estado (CIBE), melhoramentos e correcção de erros detectados no módulo de Recursos Humanos, e também nos módulos de Administração e de BackOffice.

1.3. Identificação e contextualização

O Plano Oficial de Contabilidade das Autarquias Locais (POCAL), regulamentado pelo Decreto-Lei nº 54-A/99 de 22 de Fevereiro, entrou em vigor a 1 Janeiro de 2002, constituindo uma das principais reformas da administração pública local e das contas públicas nos últimos anos. Constitui também um novo sistema contabilístico a aplicar nas autarquias locais e entidades equiparadas, visando a criação de condições para a integração consistente da contabilidade orçamental, patrimonial e custos numa contabilidade moderna, de forma a constituir um instrumento fundamental de apoio à gestão das autarquias locais e habilitar os seus respectivos órgãos de instrumentos que lhes permitam conhecer o valor total do seu património e gerir eficazmente os bens, direitos e obrigações da autarquia.

O POCAL encontra-se subdividido nos regimes simplificado e completo. O regime simplificado abrange maioritariamente as juntas de freguesia. O regime completo abrange sobretudo os municípios e empresas municipalizadas. Existem no entanto outras entidades sujeitas à aplicação do POCAL, tais como as associações de municípios, associações de freguesias de direito público, as regiões de turismo, as assembleias distritais, as áreas metropolitanas.

1.4. Motivação

O POCAL trata-se de uma importante medida no plano da gestão autárquica, que pretende dar respostas às insuficiências do anterior regime contabilístico, nomeadamente, possibilitar um conjunto de soluções em vários domínios da gestão financeira, das quais se destacam:

- A possibilidade de permitir uma gestão financeira ao interligar a contabilidade Orçamental com a contabilidade Patrimonial e com a contabilidade de Custos, o que facilita não só a avaliação da eficiência e eficácia dos serviços públicos, mas também o controlo e gestão do património autárquico;
- A disponibilidade de informação para apoiar o controlo da actividade financeira da administração local, quer pelos respectivos órgãos e cidadãos, quer ainda pelas entidades com competência legal neste domínio;
- O conhecimento integral e exacto da composição do património autárquico;
- A possibilidade de determinar correctamente, através da contabilidade de custos, o custo real de cada um dos serviços prestados, actividades e acções levadas a cabo pelos organismos sujeitos à aplicação deste novo diploma;
- A tomada de decisões estratégicas no domínio da orçamentação plurianual, face ao acompanhamento de compromissos com reflexos em anos futuros;
- A capacidade do sistema fornecer informação económica e financeira rigorosa que melhor habilite a tomada de decisões, quer ao nível da gestão corrente, quer ao nível estratégico.

1.5. Objectivos

O projecto consistiu na modelação e implementação do módulo de Gestão do Património. Este módulo permite armazenar toda a informação dos Bens, registar todas as alterações patrimoniais e amortizações, permitindo uma maior organização e transparência no património existente, permite também gerar todos os mapas definidos na portaria para apoiar a gestão dos Bens.

De seguida, foram realizados melhoramentos como a correcção de erros detectados no módulo de Recursos Humanos, no módulo de Administração e *BackOffice*. Para implementar este novo produto, a empresa criou uma nova área de negócio, constituída por uma equipa de quatro programadores e um coordenador. A aplicação foi desenvolvida nas instalações da empresa, em Lisboa, com uma carga horária de 40 horas semanais.

1.6. Instituição de Acolhimento

1.6.1 AmbiSIG – Ambiente e Sistemas de Informação Geográfica

A AmbiSIG é uma empresa de desenvolvimento de *software* e de consultadoria, e tem como objectivo a produção de soluções da mais elevada qualidade, otimizando e simplificando os processos. Actualmente conta com cerca de 60 profissionais especializados na concepção e desenvolvimento de soluções à medida.

É uma empresa com estatuto, reconhecida pela qualidade do software produzido, tendo ganho alguns prémios tais como:

Prémio das “Boas Práticas no Sector Público” atribuí duas menções honrosas aos projectos que a AmbiSIG implementou na Câmara Municipal da Horta e Maiambiente.

É uma empresa certificada pela qualidade – Norma EN ISO 9001 – para as actividades de desenvolvimento de Sistemas de Informação Geográficos e Portais, Portais Geográficos, *WorkFlow* e Gestão Documental.

O sistema da qualidade da empresa segue o modelo da norma ISO 9001:2000 ilustrado pelo esquema abaixo descrito:

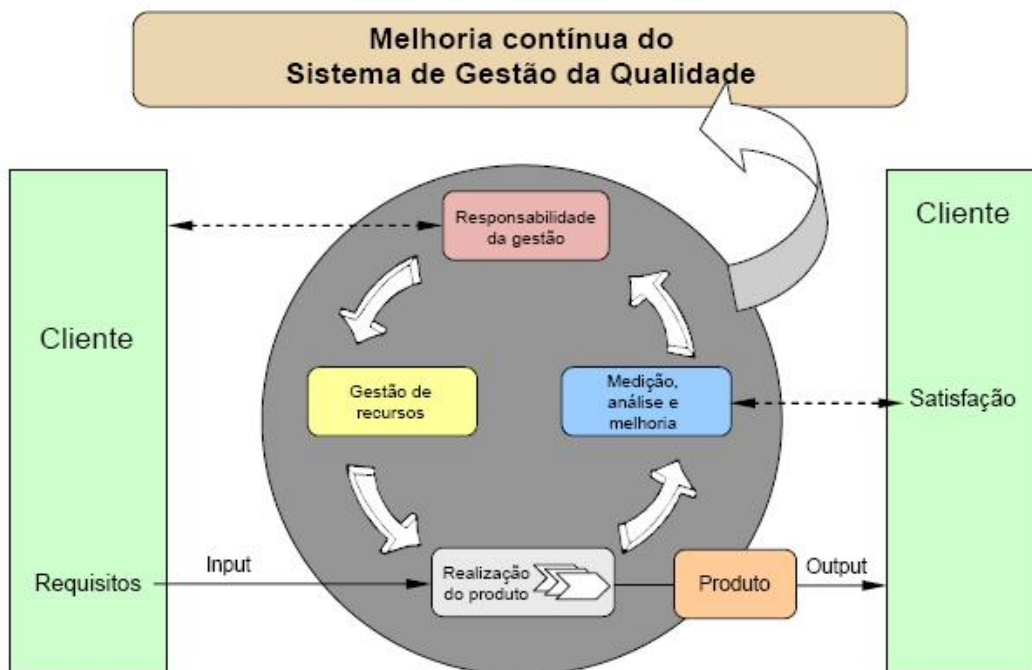


Fig. 1 - Modelo da qualidade adoptado na empresa

É, sobretudo, um sistema de gestão da qualidade orientado para a satisfação do cliente. Assenta num ciclo de melhoria contínua, que é iniciado com a identificação das necessidades e expectativas do cliente, com base nas quais, e de acordo com a estratégia adoptada, a gestão define a política da qualidade e estabelece os objectivos da mesma. O sistema de qualidade da empresa é apoiado por uma rede interna (intranet da qualidade). Esta aplicação permite a gestão e o controlo da documentação, das auditorias, das necessidades de formação, informação sobre objectivos e cálculo automático de indicadores de qualidade, suporte à implementação de acções correctivas do sistema, gestão de comunicações internas, suporte de envio de formulários de avaliação da satisfação dos clientes, e fluxos de trabalho.

A AmbiSIG é uma empresa acreditada para actividades formativas e actualmente as acções de formação surgem associadas a projectos de apoio à decisão, gestão processual e documental, submissão e tramitação de processos online e implementação de sistemas de gestão de sistemas de qualidade e ambiente, em que a formação assume um papel fulcral.

O organograma seguinte descreve a estrutura organizacional da empresa:

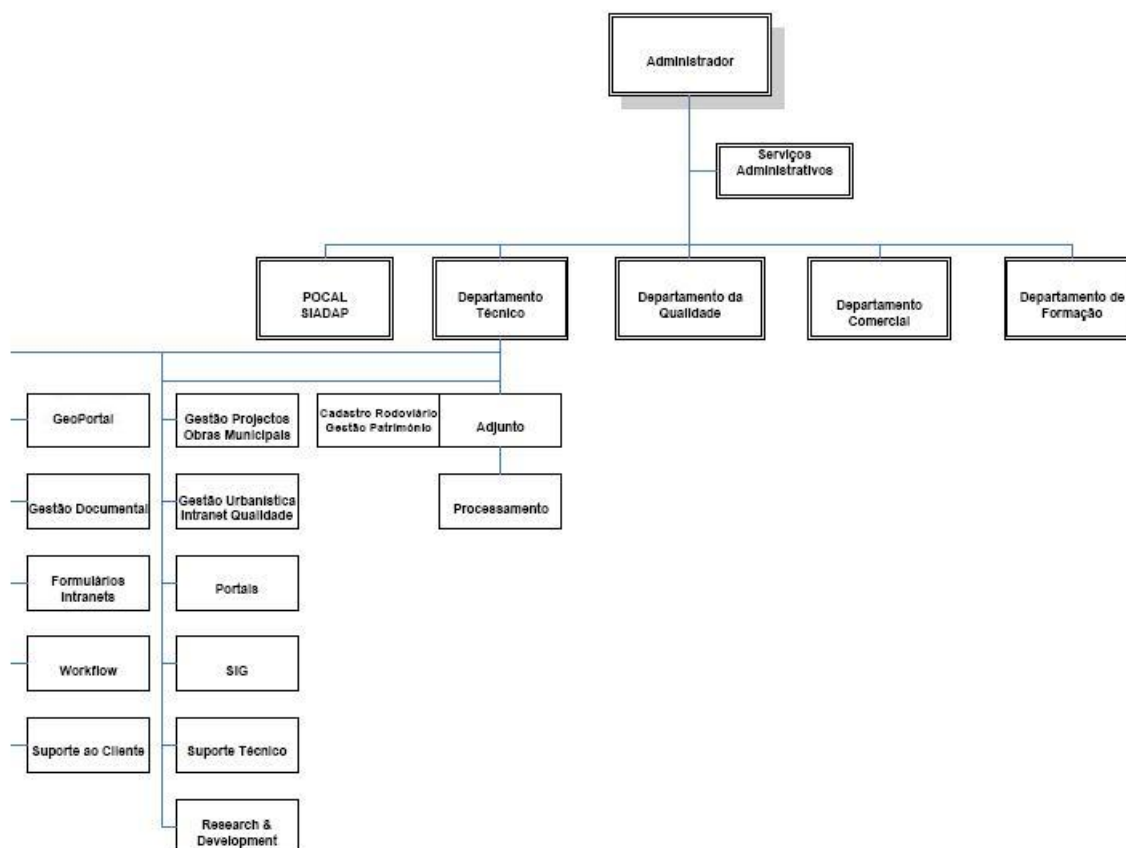


Fig. 2 - Organograma da empresa

1.7. Resumo

O aplicação eGOV.pocal foi desenvolvida de raiz e tem como objectivo complementar a oferta de produtos da AmbiSIG ao nível da Gestão Autárquica. Pretende-se com esta solução disponibilizar uma ferramenta que permita responder de uma forma simples, integrada e eficiente, às necessidades contabilísticas de prestação de contas e de gestão administrativa das juntas de freguesia e outras entidade sujeitas ao regime simplificado do POCAL, abstraindo da aplicação em si a implementação deste sistema contabilístico.

Este projecto possibilitou-me a aquisição de novos conhecimentos, nomeadamente da tecnologia Microsoft ASP.NET 2.0.

Sendo uma aplicação produzida de raiz permitiu o acompanhamento de todo o processo, desde a modelação do problema até aos respectivos melhoramentos e testes.

O projecto consistiu na modelação e implementação do módulo de Gestão Património. Para implementar este projecto foi constituída uma equipa de quatro programadores e um gestor. Todo o projecto foi implementado nas instalações da empresa, com uma carga horária de 40 horas semanais.

2. Planeamento do Projecto

2.1. Análise e justificação da viabilidade do projecto

Este projecto visou a implementação do sistema POCAL e pretende-se que a solução final permita responder de uma forma simplificada e automatize todos os processos inerentes, num ambiente web e descentralizado. Esta solução vai permitir munir as autarquias, e todos os órgãos afectos ao regime simplificado do POCAL, de uma ferramenta que permita conhecer o valor total dos bens e possa geri-los eficazmente, assim como os direitos e obrigações das autarquias. Esta aplicação é facilmente parametrizável à medida das necessidades dos clientes. Este projecto gera a possibilidade de se ter uma gestão financeira integrada, a contabilidade orçamental com a contabilidade patrimonial e com a contabilidade de custos, o que facilita não só a avaliação da eficiência e eficácia dos serviços público, mas também o controlo e gestão do património autárquico. Todas as fichas e mapas são gerados em formato PDF, o que, para além da possibilidade de impressão, simplifica os processos de distribuição por *email* ou de arquivo digital. Esta plataforma é totalmente baseada na mais recente tecnologia web, a solução é instalada num só servidor IIS e os utilizadores apenas necessitam de ter instalado um browser nos seus computadores. É um sistema independente do tipo base de dados, usando a mais recente tecnologia Microsoft ASP.NET 2.0. Não existem investimentos adicionais ao nível de hardware, visto poder ser acedida num browser. É uma gestão simplificada sem necessidade de recorrer à utilização de conhecimentos técnicos, há uma redução entre os ciclos de trabalho, e o licenciamento é independente do número de utilizadores. Existe um enorme leque de clientes o que faz com que esta solução seja ainda mais viável (só as juntas de freguesia são mais de 4000).

2.2. Definição da equipa para a implementação do projecto

Para implementar esta aplicação foi constituída uma equipa de quatro programadores e um coordenador. O coordenador foi o responsável pela selecção dos candidatos e pela coordenação de acções estratégicas, tecnológicas, controlo de qualidade do projecto e processos de concepção. Todos os elementos que fazem parte desta equipa desenvolveram as suas actividades no mesmo espaço de trabalho, o que foi um ponto positivo, pois permitiu facultar o enriquecimento pessoal através de troca de ideias e também fomentou o espírito de equipa e a partilha de conhecimentos.

2.3. Definição e apresentação do plano e das estratégias para implementação do sistema

Antes da aproximação prática existe todo um conjunto de normas pelo qual qualquer projecto se deve orientar. O levantamento e análise de requisitos, pesquisa, planeamento e definição de arquitectura são as fases que devem ser focadas.

Numa fase inicial, o meu trabalho passou por conhecer o funcionamento da empresa, as políticas que a empresa impõe como, por exemplo, as convenções de escrita de código e de modelação de dados. Tive formação na *Framework* da AmbiSIG que funciona sobre a *Framework. NET*, e também sobre o manual de qualidade que descreve o sistema de qualidade da empresa.

Houve inicialmente uma maior dedicação da minha parte na aquisição de conhecimentos do software que iria ser usado na produção da aplicação. Para uma melhor compreensão do projecto comecei pela leitura aprofundada de legislação relativa ao Cadastro e Inventário dos Bens do Estado. Foi-me ainda facultada uma aplicação anteriormente produzida pela empresa, para gestão de património, desenvolvida em Microsoft Visual Basic 6.0. A grande limitação desta aplicação é ser feita em *Windows forms* o que limitava o acesso à aplicação e também não estava actualizada em relação ao Decreto-Lei que se encontra em vigor. Esta aplicação serviu para retirar várias ideias nomeadamente, ao nível da interface, na disposição dos elementos, na estrutura e organização do menu, e também o modelo de dados, foi o ponto de partida para a construção deste novo modelo de dados. O prazo de entrega da aplicação foi de nove meses.

2.4. Metodologia e Calendarização

Independentemente da metodologia escolhida para a construção de um produto de *software*, a mesma deve possuir uma semântica bem definida, para que qualquer gestor interprete e seja capaz de entender sem ambiguidades os propósitos do sistema.

O termo metodologia, apesar de ser muito usado, não possui uma única definição. Em geral, entende-se como metodologia uma série de passos e procedimentos que devem ser seguidos no desenvolvimento de um sistema de informação (Yourdon, 1995).

De acordo com Avison e Fitzgerald (1997), é o conjunto formado por procedimentos, técnicas, ferramentas e documentação que auxilia os gestores na implementação do sistema.

As actividades de uma metodologia podem e devem ser adaptadas a cada tipo de projecto, de acordo com o ambiente tecnológico, complexidade e limitações de prazo ou natureza económica.

O uso de uma metodologia significa, assim, que se está a seguir um método já testado, sólido e consistente que proporcionará uma melhor qualidade e eficiência na solução final.

2.4.1 Metodologia

A metodologia usada na produção do POCAL foi o modelo cascata. Este modelo desenvolvido no final da década de 1960 é, ainda hoje, a abordagem mais usada na indústria de *software*. Neste modelo as tarefas são executadas sequencialmente de forma que uma tarefa só inicia quando a tarefa anterior terminar (fig.3).

A empresa adopta uma metodologia centrada nas necessidades do cliente. Sempre que haja necessidade devem ser pedidos esclarecimentos ao cliente, para garantir que os seus requisitos sejam claramente entendidos. Com esta abordagem, pretende-se uma maior interacção com o cliente, o que reflecte numa maior satisfação do cliente, e na qualidade do *software* produzido.

Neste projecto, visto ser uma aposta da empresa no desenvolvimento de um novo produto, inicialmente não existia nenhum cliente associado, sendo assim impossível existir interacção com este para recolha de requisitos. Posteriormente, depois de implementada uma solução parcial e ser apresentada a potenciais clientes, foi possível perceber se a aplicação estava a ter o impacto desejado a partir do *feedback* e interesse demonstrado na apresentação e efectuar um conjunto significativo de melhoramentos.

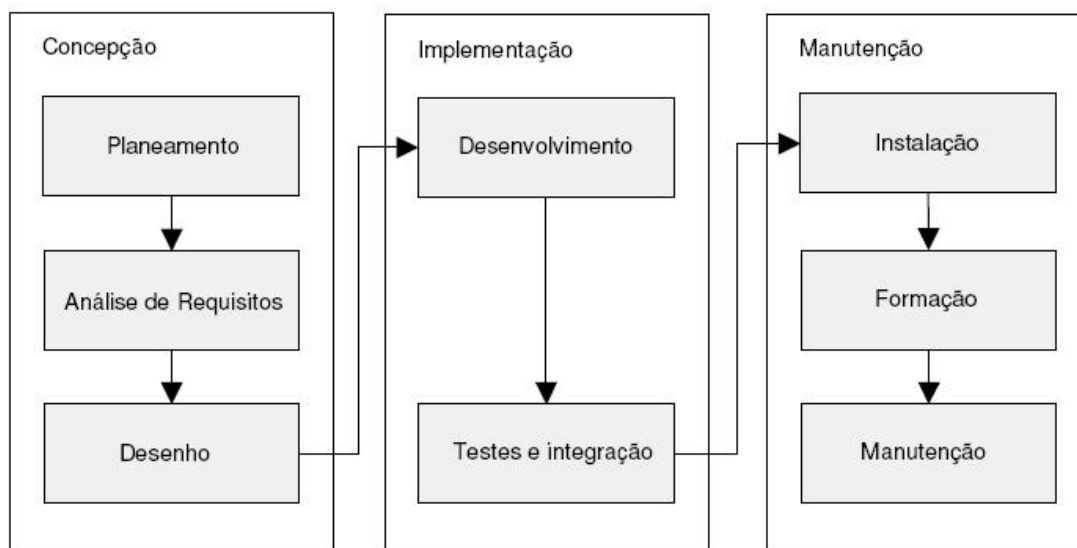


Fig. 3 - Fases e Tarefas do processo de desenvolvimento de software

- **Planeamento:**

O planeamento corresponde a uma identificação geral das necessidades, identificação e selecção de alternativas e definição de plano de trabalho. A grande preocupação desta fase é, a partir de um levantamento de alto nível das necessidades, que seja possível elaborar um plano do projecto a executar nas fases subsequentes, com a identificação das actividades, recursos, prazos e custos.

- **Análise dos Requisitos:**

A tarefa de análise efectua o estudo detalhado do domínio do problema e culmina na elaboração de um documento onde os requisitos funcionais da solução a implementar são enumerados. Um requisito é uma funcionalidade que o sistema deverá possuir. Para os identificar adequadamente é aplicado um conjunto de técnicas de modo a obter a percepção detalhada daquilo que o sistema deverá ser. Contrariamente ao que se poderia julgar em primeira instância, definir e descrever os requisitos de um sistema não é tarefa fácil. Esta actividade pode incluir a realização de reuniões com os interessados, a elaboração de questionários, funcionamento do dia-a-dia, a recolha e análise de documentos, a elaboração de pequenos protótipos que permitam validar mais facilmente a percepção. O objectivo desta tarefa é expressar sem ambiguidades o que o sistema deve fazer e não como fazer o sistema. No Documento de Risco são identificados os riscos que podem comprometer o sucesso do projecto, é analisada a gravidade de cada risco e descrevem-se as acções que minimizam cada risco. O Planeamento é efectuado com base nos requisitos do cliente.

A Especificação Funcional, o Documento de Risco e o Planeamento, são discutidos e avaliados conjuntamente com o cliente para aprovação. O gestor do projecto é o responsável pela elaboração e revisão destes documentos.

- **Desenho:**

O desenho procede à especificação formal das características que a implementação do sistema deverá apresentar, assim como a realização de determinadas optimizações consoante as características da infra-estrutura de objectos, topologias de redes etc. É também nesta tarefa que deve ser completamente definido o ambiente e as linguagens a utilizar no desenvolvimento. A AmbiSIG utiliza maioritariamente no desenvolvimento das suas aplicações a tecnologia *Microsoft*. Este projecto não foi excepção.

- **Desenvolvimento:**

A tarefa de desenvolvimento inclui todas as actividades relativas de implementação do sistema propriamente dito, ou seja, que estão relacionados com a concretização do modelo de desenho produzido na tarefa anterior. Os diversos componentes da aplicação são codificados e testados de forma isolada garantido assim uma correcção interna (testes unitários). Sempre que necessário devem ser pedidos esclarecimentos com o cliente para clarificar algum requisito incompreendido.

- **Testes e Integração:**

Para além dos testes unitários, esta tarefa destina-se à realização dos restantes testes. O objectivo é avaliar a adequada correcção e funcionamento de todos os componentes do sistema. A verificação consiste na confirmação que a codificação (implementação) do sistema está conforme (correcta) com a especificação técnica produzida na fase do desenho.

- **Instalação:**

Esta tarefa consiste na preparação e instalação do sistema na infra-estrutura computacional do cliente e envolve um conjunto de tarefas muitas vezes esquecidas na altura da preparação do plano de projecto:

- Instalação dos sistemas de suporte técnico;
- Configuração e parametrização do sistema implementado;
- Definição de perfis, de utilizadores, e de níveis de segurança.

Caso exista já um sistema em funcionamento é necessário preparar a migração para o novo sistema, por exemplo as fichas de cadastros ou a estrutura orgânica de uma autarquia.

- **Formação:**

Uma das tarefas fundamentais para o sucesso do projecto é a formação dos utilizadores no novo sistema. Esta fase deve ser acompanhada com documentação.

- **Manutenção:**

Durante a vida útil de qualquer sistema de *software* são detectados problemas que não são devidamente verificados durante a fase de implementação. Surgem também inúmeras solicitações internas e externas relativamente a pedidos de alteração de requisitos que exigem a elaboração de novas versões / actualizações do *software*. O objectivo da manutenção é garantir que a ocorrência de alguma destas situações seja convenientemente tratada.

Todos os produtos produzidos pela AmbiSIG têm uma garantia de dois anos.

2.4.2 Calendarização

O planeamento geral é apresentado na tabela seguinte:

Etapas	Duração	Data Início	Data Fim
0 – Estudo sobre os processos	1 Semana	01-Ago-2006	7-Ago-2006
1 - Análise de Requisitos	2 Semanas	08-Ago-2006	20-Ago-2006
2 - Modelo de Dados	2 Semanas	20-Ago-2006	10-Set-2006
3 – Desenvolvimento Módulo de Gestão Património	20 Semanas	11-Set-2006	01-Mar-2007
4 – Verificação de Conformidade	4 Semanas	01-Mar-2007	01-Abr-2007
5 – Melhoramentos / Novas funcionalidades	4 Semanas	01-Abr-2007	01-Mai-2007
Elaboração do Relatório de MEI		Realizado paralelamente ao projecto	

Tabela 1 - Planeamento dos principais módulos do projecto

Serão necessárias cinco etapas a fim de executar este projecto. Essas etapas encontram-se descritas na tabela 1. O plano para a execução das actividades contidas nestas etapas é apresentado a seguir:

- **Etapa 0 – Estudo sobre os processos**

Objectivo: Esta etapa visa adquirir o conhecimento necessário sobre os processos e políticas que estão implementados na empresa. A aquisição deste conhecimento passa pela formação sobre os processos, pela leitura de manuais existentes na empresa, mais especificamente o Manual da Qualidade que descreve o Sistema da Qualidade, desenvolvido segundo a norma NP EN ISO 9001, aplicável ao desenvolvimento de *software*. Este documento define os objectivos e compromissos assumidos pela empresa no domínio da qualidade, constituindo assim um documento de referência para a empresa e clientes. Este documento pretende traduzir as práticas e procedimentos adoptados pela empresa.

Atividade 0	
Descrição	estudo das práticas e procedimentos adoptados pela empresa
Entrada	manual de qualidade e outros documentos disponibilizados pela empresa, formação sobre o manual de qualidade e Formação sobre a FrameWork
Metodologia	o estudo será feito através da leitura dos manuais e na formação proposta
Saída	aquisição dos processos adotados pela AmbiSIG
Duração	1 semana

Tabela 2 - Aquisição dos conhecimentos dos processos adoptados pela empresa

- **Etapa 1 – Análise de Requisitos**

Objectivo: Esta etapa visa a análise e recolha de requisitos.

Atividade 1	
Descrição	aquisição dos Requisitos
Entrada	livro POCAL Diário da República, legislação relativa ao POCAL artigos sobre o POCAL
Metodologia	o estudo será feito através da leitura de artigos, livros e sites
Saída	manual da Especificação funcional
Duração	2 semanas

Tabela 3 - Aquisição dos requisitos POCAL

Nesta fase pretende-se que seja extraído dos recursos disponíveis os requisitos. Esta tarefa permitiu uma melhor compreensão e percepção do projecto. Foi produzido um documento, onde constam os requisitos, que foi posteriormente validado pelo coordenador de projecto.

Depois de ler os documentos que me foram facultados comecei a entender melhor o trabalho que iria realizar, denotei que a informação relativamente ao cadastro do bem é muito permonorizada, cada bem é identificado por um identificador único sequencial, também verifiquei que existem três tipos de categorias de bens (Móveis, Imóveis e Veículos), que a aplicação teria que suportar todos os tipos de variações patrimoniais que possam ocorrer nos Bens e também que existem mapas oficiais que são gerados a partir da informação dos Bens.

- **Etapa 2 – Modelo de Dados**

Objectivo: Esta fase visa a elaboração do modelo de dados que suporte os requisitos enunciados na especificação funcional.

Atividade	2
Descrição	Elaboração do Modelo de Dados
Entrada	livro POCAL Diário da República, Legislação relativa ao POCAL artigos sobre o POCAL manual especificação funcional
Trabalho Realizado	O estudo será feito através da leitura de artigos, livros e sites
Saída	modelo de dados
Duração	2 semana

Tabela 4 - Elaboração do modelo de dados

O modelo de dados final encontra-se no anexo 3 – Modelo de Dados. Este modelo, durante o projecto, levou sucessivos refinamentos, pela razão de não existir um cliente associado que validasse inicialmente os requisitos. Mais uma vez observou-se que a interacção entre cliente é muito importante para garantir que os requisitos são claramente entendidos e definidos à partida.

Nesta fase comecei por migrar da antiga aplicação gestão de património o modelo de dados, este estava implementado numa base de dados *Oracle* e pretendia mudar para *SQL Server*. Este modelo de dados foi alterado de forma a suportar o novo Decreto-Lei. A empresa tem uma política na convenção do nome das tabelas que é o seguinte: as tabelas de metadados começam sempre por ‘md’, estas tabelas auxiliares são usadas para armazenar informação de todas as tabelas e respectivos campos, esta metodologia permite a atribuição de permissões de uma forma fácil e controlada, as restantes tabelas devem começar pelo nome da aplicação seguido do módulo e depois o nome da tabela, no meu caso, todas as tabelas devem começar por ‘PCCIBE’ , ‘PC’ de POCAL e ‘CIBE’ de Cadastro e Inventário de Bens de Estado.

- **Etapa 3 – Desenvolvimento Módulo de Gestão Património**

Objectivo: Nesta fase é executado o projecto de acordo com os requisitos funcionais.

Atividade	3
Descrição	Implementação do Módulo Cadastro de Património
Entrada	Manual da Especificação funcional
Trabalho Realizado	O Trabalho consiste na produção de todo o módulo de Cadastro dos bens, uma componente de parametrização e Mapas que estão abrangidos pela lei
Saída	A solução
Duração	5 meses

Tabela 5 - Implementação do Módulo de Cadastro

Nesta fase tive o primeiro contacto com ASP.NET, logo, apareceram as dificuldades habituais, que ocorrem aquando a aprendizagem de uma nova linguagem, tais como, entender a estrutura de uma aplicação ASP.NET, a configuração de aplicações ASP.NET, a construção dos formulários, as convenções de escrita de código adoptados pela empresa, o conceito de *master page*, *profiles*, *sessions*, *cookies*, *datasource*, *webservice*, como era feita a gestão dos ficheiros, como são feitos os relatórios, como se gerem as permissões na aplicação, foram muitas as dificuldades que ocorreram no início, mas existiu muito apoio da equipa, que estava sempre pronta e disponível para me auxiliar em caso de dúvidas.

A empresa disponibilizou um *add-in* para a construção de formulários simples. Este *add-in* foi muito importante para perceber como é que os vários ficheiros deveriam estar organizados, pois gera automaticamente de uma tabela da base de dados um formulário, mas depois demonstrou-se ser muito limitado pois não funcionava para formulários que referenciam mais que uma tabela.

- **Etapa 4 – Verificação de Conformidade**

Objectivo: Nesta fase pretende-se efectuar a verificação final da conformidade do trabalho produzido face à especificação funcional e ao planeamento. As situações de não conformidade são registadas pelo suporte ao cliente, comunicando-as ao coordenador de projecto que define um responsável e prazo para a correcção. Após correcção, o suporte ao cliente deverá efectuar a verificação para assegurar a conformidade.

Atividade	4
Descrição	Verificação de Conformidades
Entrada	Manual da Especificação funcional Solução desenvolvida
Trabalho Realizado	O Trabalho consiste verificação se o produto está em conformidade com os requisitos funcionais
Saída	Documento com as não conformidades
Duração	1 mês

Tabela 6 - Verificação de conformidades

Depois de o coordenador definir um responsável, cabia a um elemento da equipa corrigir o erro, os erros podiam ser desde alterar a *label* de um controlo, adicionar novos controlos ou corrigir um mapa, o preenchimento das *dropdownlist*, a interface, etc. ...

- **Etapa 5 – Melhoramentos / Novas Funcionalidades**

Objectivo: Esta fase tem como objectivo realizar melhoramentos e novas funcionalidades que foram discutidas aquando da apresentação da solução a potenciais clientes.

Atividade	5
Descrição	Melhoramentos a aplicação e implementação de novas funcionalidades
Entrada	<i>Feedback</i> do cliente em apresentação realizadas
Trabalho Realizado	O trabalho consiste em implementar ideias que enriqueçam a solução
Saída	Novas funcionalidades
Duração	1 mês

Tabela 7 - Melhoramentos e implementação de novas funcionalidades

Nesta fase o objectivo consistiu na realização de apresentações a clientes, essas apresentações normalmente são realizadas pelo coordenador do projecto ou então pelo Departamento Comercial. Durante as apresentações normalmente o cliente opinava sobre a aplicação, essas ideias eram anotadas e depois o coordenador decidia se realmente a opinião era importante para o enriquecimento da aplicação, e em caso afirmativo, nomeava uma pessoa para realizar essa alteração.

Inserir novas funcionalidades na aplicação era muito importante, pois ao enriquecer a solução melhoramos a qualidade do próprio software o que aumenta a satisfação do cliente.

3. Arquitectura e ferramentas Utilizadas

3.1. Arquitectura

A arquitectura usada para a construção da solução é baseada na cliente-servidor e num modelo de três camadas (Fig 4). Este modelo separa a interface com o utilizador (*browser*) do nível das aplicações lógicas (servidor) e da base de dados.

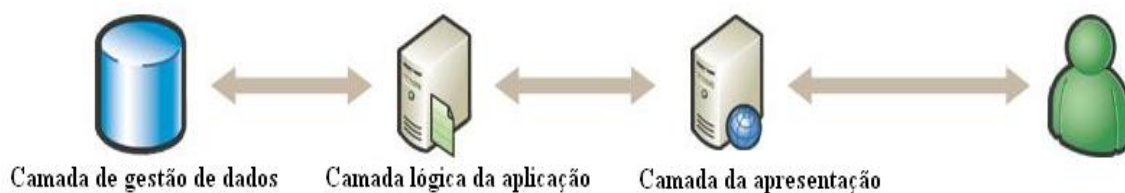


Fig. 4 - Arquitectura 3 camadas

A **Presentation Layer** (camada de apresentação), também designada *client-tier*, corresponde à interface do utilizador com o sistema, assumindo, geralmente, a forma de um *browser* web. Aqui são prestados os mais diversos serviços ao utilizador; salienta-se a visualização e manipulação de dados, e permite ao utilizador usar o sistema.

A **Business Rules / Logic** (camada da lógica aplicação), também referida como *middle-tier* ou *server-tier*, é a camada que estabelece a ligação entre a camada de apresentação e a camada de gestão de dados. Nesta camada são definidas as regras e políticas de gestão, como permissões e restrições de acesso que devem ser asseguradas pela aplicação e de que forma os recursos devem ser utilizados.

A **Data Management** (camada de gestão de dados), também intitulada *database-tier*, corresponde à base de dados propriamente dita. Esta camada é responsável pela gestão

funcional dos dados assegurando os aspectos que se relacionam com a segurança (protecção contra acessos não autorizados) e integridade dos dados (controlo de concorrência e recuperação / tolerância a falhas).

O processo de comunicação neste modelo funciona da seguinte forma:

- Antes de mais, o cliente acede à informação através de páginas ASPX que vão sendo apresentadas no *browser*. Neste primeiro passo o utilizador faz um pedido que é encaminhado pelo *browser* para o servidor da aplicação.
- Após o pedido ter recebido pelo servidor de aplicações, este é encaminhado para a base de dados.
- O servidor de base de dados recebe o pedido, reencaminha-o ao servidor da aplicação, e vai executar o respectivo código SQL que permite obter a resposta ao pedido efectuado pelo cliente. Após executado este código, o servidor de base de dados devolve a resposta ao servidor da aplicação.
- Neste passo, o servidor de aplicação fica encarregue de devolver uma página no formato ASPX que irá conter a informação que a base de dados disponibilizou e que constitui a resposta ao solicitado pelo *browser*.

O uso desta arquitectura tem como vantagem o aumento de desempenho porque, são feitos pequenos pedidos de dados ao servidor de base de dados, ou seja, isto significa que menos dados são enviados através da rede. A aplicação pode não estar centralizada pelo que não obriga que haja um SGBD específico. Um ponto desfavorável é que nesta arquitectura nem sempre é fácil / óbvio separar a lógica das camadas: interface com o utilizador, gestão dos processos e dos dados.

A solução do *Microsoft* para a internet recorre a este modelo três camadas, em termos de páginas temos as ASPX (páginas de apresentação), no servidor de aplicação o Serviços de Informação Internet (IIS) e na base de dados o SQL Server.

3.2. Ferramentas Utilizadas

Sendo a AmbiSIG uma empresa certificada pela *Microsoft*, é natural que a tecnologia utilizada predominantemente no desenvolvimento seja *Microsoft*. O *software* que foi instalado para desenvolver a aplicação foi:

3.2.1 Linguagens de Programação

- **Visual Studio 2005:** É o ambiente de desenvolvimento de aplicações *web*. O *Visual Studio* oferece uma variedade de ferramentas que proporciona vários benefícios para equipa como, por exemplo, o aumento da produtividade.
- **Asp.net 2.0:** O ASP.NET é uma poderosa ferramenta de programação que permite a criação de aplicação *web* dinâmicas. O ASP.NET é baseado na framework .NET herdando todas as suas características, por isso, como qualquer aplicação .NET, as aplicações para esta plataforma podem ser escritas em várias linguagens como C# e VB.NET. A linguagem utilizada foi o VB.NET.
- **IIS (Internet Information Service):** É uma ferramenta da Microsoft que permite a execução de aplicações baseadas na internet que correm num servidor windows.
- **FrameWork AmbiSIG:** É uma camada que funciona sobre a framework .NET 2.0 que possibilita, entre outras coisas, realizar uma aplicação independente da base de dados, permite suportar diferentes idiomas, segurança ao nível de tabelas, classes que geram SQL e permitem ao programador abstrair-se da base de dados utilizada e muito mais.
- **Add-in AmbiSIG:** Esta aplicação criada pela empresa permite gerar páginas a partir de tabelas, diminuindo muito o tempo de criação das páginas.
- **Atlas:** O “atlas” é uma *framework* gratuita para a criação de aplicações *web* mais ricas, interactivas baseadas em padrão e com alto poder de personalização. Já integra as funcionalidades do AJAX, fornece uma framework rica do lado do cliente,

permitindo realizar actualizações parciais sem que a página onde estão incluídas tenha de ser actualizada na sua totalidade.

- **JavaScript:** Permite fazer operações do lado do cliente sem realizar a actualização total da página.
- **Crystal Reports:** É uma ferramenta que permite criar mapas com base em queries sobre a Base de Dados. Esta ferramenta disponibiliza ainda funcionalidades interessantes do ponto de vista do utilizador, tais como a exportação dos dados para MS Word ou Excel, entre outros formatos.

3.2.2 Base de Dados

- **Microsoft SQL Server 2005:** No ambiente de produção foi adoptado o uso deste sistema de gestão de base de dados, por ser tecnologia Microsoft, mas é de realçar que a aplicação é independente do SGBD devido à framework da AmbiSIG.

3.2.3 Outras Ferramentas

- **Visual Source Safe:** O VSS é uma ferramenta de colaboração para programadores de aplicações. Com esta ferramenta é possível, dentro da equipa de desenvolvimento, controlar os acessos e versões de cada componente da aplicação. A grande vantagem de usar este software e não outro semelhante existente no mercado prende-se com a integração com os produtos Microsoft. Esta foi a ferramenta adoptada para controlar o código fonte do projecto.
- **CSS:** Cascading Style sheets, ou simplesmente CSS, uma linguagem para definir o estilo de apresentação das páginas (cores, formatação do texto, posicionamento de texto, etc.)
- **MSDN Library:** É um recurso essencial para quem está a usar ferramentas, produtos e tecnologia Microsoft. É possível encontrar informação interessante, incluindo pedaços de código, documentação, artigos técnicos e manuais de referência.

A aplicação *web* eGOV.pocal foi desenvolvida em ASP.NET sobre a ferramenta *Microsoft Visual Studio 2005*. O ASP.NET é uma poderosa ferramenta de programação que permite a criação de aplicações *web* dinâmicas, faz parte integrante da *framework* .NET da *Microsoft*. A versão utilizada da *framework* .NET foi a 2.0. Juntamente com o ASP.NET foram utilizadas outras linguagens de apoio que são suportadas pela *framework* .NET, nomeadamente HTML, VB.NET, ATLAS, JavaScript. Uma aplicação *web* implica a configuração do servidor de *web*, neste caso, o IIS (*Internet Information Services*), bem como o DNS (*Domain Name Server*). O IIS é uma ferramenta da *Microsoft* que permite a execução de aplicações baseadas na internet que ocorrem num servidor *Windows*. O DNS é um serviço internet que traduz os nomes de domínios em endereços IP.

Para a gestão dos ficheiros, o *source-safe* a meu ver é uma ferramenta imprescindível no mercado de trabalho, permite ter um historial de versões, saber quem realizou cada alteração ao código e quando.

Para a geração de todos os mapas foi usado o *Crystal Reports*, esta aplicação é uma poderosa ferramenta para a geração de relatórios e oferece uma forma produtiva e rápida de criar e integrar relatórios interactivos com qualidade para ser apresentados aos utilizadores finais, os relatórios podem ser exportados em vários formatos, PDF, *excel* e *word*. O *Crystal Reports* pode ser utilizado pela maioria das linguagens de programação existentes no mercado em associação com base de dados como *Oracle*, *Microsoft SQL Server*, entre outros.

O AJAX (*Asynchronous JavaScript and XML*) usa blocos de código escritos em *javascript* efectuando chamadas a aplicações *web* via protocolo XMLHTTP. Este protocolo não é novo, está actualmente presente na maioria das aplicações *web*, e é ele que permite efectuar chamadas assíncronas a um servidor *web*. Implementar e usar o AJAX não é tão simples como parece, requer o conhecimento de *javascript*, DHTML e protocolos XMLHTTP. É aqui que entra o *atlas*, esta tecnologia foi criada pela *Microsoft* com o objectivo de encapsular as funcionalidades do AJAX e tornar mais fácil sua implementação e utilização. A grande vantagem de utilizar o *atlas* para quem desenvolve aplicações *web* com ASP.NET, é ela encontrar-se totalmente integrada com o ASP.NET 2.0 presente no *Visual Studio 2005*. Além disso, o *Atlas* disponibiliza recursos que não estão presentes no AJAX, entre eles, extensão de novas funcionalidades nos controlos ASP.NET, por exemplo, na *textbox*, a capacidade de completar o texto à medida que se vai digitando, o carregamento de dados em cascata no controlo *dropdownlist*, mas a mais importante é o carregamento parcial das páginas, evitando assim que a página seja carregada na sua

totalidade em cada clique do utilizador. Isto transmite um efeito muito agradável, pois o modelo *web* passa a ter o funcionamento similar ao modelo *windows*.

A AmbiSIG tem desenvolvido, ao longo do tempo, uma *framework* própria que permite aos programadores uma rápida e fácil reutilização de módulos comuns a todas as aplicações que dispõe. A *framework* da AmbiSIG tem um sistema de permissões baseado em módulos, contextos e tabelas. A utilização de metadados é de extrema importância para que se possa ter o controlo de uma aplicação. Todas as tabelas e respectivos campos são guardados em tabelas auxiliares de metadados. Esta metodologia permite a atribuição de permissões de uma forma fácil e controlada. A *framework* da AmbiSIG encapsula todas as queries que são feitas à base de dados, possibilitando o modelo de dados ser construído em várias linguagens, por exemplo, *Oracle*, *SQL*, *MySQL* e a abstração do programador.

4. Trabalho Realizado

4.1. POCAL

O desenvolvimento da solução implica o conhecimento da *framework* de desenvolvimento da AmbiSIG, bem como das suas potencialidades. Para isso foi necessário formação das políticas adoptadas pela empresa de forma a existir uma mais fácil integração.

Existiu uma maior dedicação da minha parte na aquisição de conhecimentos do *software* utilizado dentro da empresa (*VB.NET, VSS, Atlas, Crystal Reports*).

Dentro da equipa denotei uma enorme entre-ajuda o que possibilitou ultrapassar mais facilmente os problemas e dúvidas que iam ocorrendo.

Este projecto foi desenvolvido em ASP.NET e pretendia-se que se implementasse o POCAL num ambiente *web*. O meu trabalho passou pela implementação do Módulo de Cadastro de Inventário de Bens de Estado - pretendia-se que com esta aplicação fosse possível o registo e manutenção da informação dos bens patrimoniais, as suas variações patrimoniais, facilmente parametrizável e com uma interface simples e intuitiva.

Antes de começar a programar foi necessário perceber o que realmente tinha que implementar, assim comecei pela leitura aprofundada da Legislação aplicável e de outros documentos de apoio. A partir daí foi criado um modelo de dados que suportasse o módulo Cadastro de Património, que depois iria interligar com os outros.

4.2. Modelo de Dados

O modelo de dados implementado para esta solução encontra-se em anexo. Este modelo, ao longo da produção da solução, teve sucessivos refinamentos e passou por diversas validações, para suportar o módulo de Gestão de Património. No modelo de dados existem tabelas, intituladas por metadados, estas tabelas armazenam informação genéricas em todas as aplicações, por exemplo, informação sobre a estrutura da aplicação (módulos, opções de módulos e contextos), permissões, tabelas, campos, utilizadores, etc.

4.3. Trabalho Desenvolvido

O Visual Studio fornece os recursos necessários para a criação de sites ASP.NET. O ASP.NET é um modelo de programação semelhante à criação de aplicações windows forms. Em web forms, o código é executado no servidor para gerar a página web, depois é enviada e exibida no browser, quando a interface é apresentada ao utilizador a classe de página do servidor já foi descarregada.

As dificuldades que ocorreram foram várias: perceber o modelo de dados dos metadados, as políticas e convenções adoptadas pela empresa, como deveria estar organizado o código no *Visual Studio*, o funcionamento da *framework* ASP.NET e da AmbiSIG, criação dos mapas com *Crystal Reports*, ligação do ASP com *Crystal reports*, etc.

Para a construção da aplicação foi necessário aprender várias conceitos, por exemplo:

- **Layout da página**

A organização e disposição dos controlos é muito importante. Não basta adicionar os controlos na página, é necessário existir alguma forma para controlar os alinhamentos dos controlos, esse organização é consiga recorrendo a CSS, uma linguagem de estilo utilizada na apresentação de páginas *web*.

- **Configuração da Aplicação**

A configuração de uma aplicação ASP é feita a partir de um ficheiro XML *web.config*. Este ficheiro contém todo o tipo de informação, desde as *strings* de conexão, páginas de erro, modo de compilação, *assemblies*. É possível também armazenar valores e parâmetros que sejam comuns em toda a aplicação.

- **Manutenção do Estado**

O armazenamento de informação é feito pela variáveis de sessão. Estas variáveis são utilizadas para armazenar informação que pode ser usada ao longo da aplicação sem que se perca a informação. Funcionam como variáveis globais, e é a forma mais fácil de passar a informação entre as páginas. Na aplicação é utilizada para armazenar o estado de controlos e informação sobre as permissões e dados do utilizador.

- **Master Pages**

A *master page* é um novo conceito introduzido pela ASP 2.0, ela permite construir um *template* que será aplicado a uma ou mais páginas. Qualquer página criada pode herdar uma *master page*. Com esse recurso podemos criar uma página padrão que será usada em toda a aplicação.

- **Autenticação**

ASP.NET 2.0 apresenta um conjunto de novos controlos que permitem realizar as principais operações relacionadas com a gestão de utilizadores. Na aplicação do POCAL foi utilizado um desses controlos, o *asp:login*. Este controlo efectua a validação das credenciais introduzidas por um utilizador. A propriedade *MembershipProvider* permite-nos definir o *provider* usado na operação de validação de credenciais. Foi criado um novo *provider* para especificar onde seria feita a verificação de credenciais de um utilizador.

O trabalho desenvolvido passou por :

- **Implementar o módulo de cadastro e inventário de gestão de património**

O módulo de Gestão de Património assenta em três sub-módulos: o cadastro onde se registam os Bens; uma componente de parametrização onde são definidos parâmetros genéricos; e os mapas, para impressão dos relatórios.

O menu é carregado a partir de *scripts* que preenchem as tabelas de metadados *mdApp*, *mdAppModule* e *mdAppModuleOption*. O carregamento é dinâmico pois é baseado nas permissões de um determinado utilizador.

Para a construção das páginas foi usado o *add-in* da AmbiSIG. Esta aplicação permite gerar a partir de uma tabela, três ficheiros com a seguinte terminação: *listview*, *formview* e *datamanager*.

O ficheiro *listview* contém normalmente um controlo *gridview* que apresenta os dados da tabela, e campos de pesquisa / filtragem. Cada registo da *gridview* tem um campo com o um *link* para o *formview* passando por *querystring*, o *id* do registo seleccionado. O carregamento dos controlos do *formview* é feito pela *framework*, que percorre todos os campos da *page* e preenche com o valor correcto. As *labels* também são preenchidas a partir dos metadados, pela tabela *mdData* e *mdDataField*, com base no campo *caption*. O carregamento das *labels* foi feito por *script*. O ficheiro *datamanager* serve para armazenar

todas as queries, rotinas e cálculos necessários a determinado formulário. Cada formulário tem um ficheiro *datamanager* associado.

- **Melhoramentos e correcções detectados no módulo de Recursos Humanos:**

Face à saída de um colega da equipa, fiquei encarregue de testar as rotinas dos cálculos do módulo de Recursos Humanos. Este trabalho passou por validar essas rotinas e verificar se o cálculo estava a ser feito da forma correcta (e corrigir os erros detectados).

- **Implementação de módulo de administração – definição de permissões**

A atribuição das permissões era feita de forma individual não permitindo o carregamento em bloco de permissões associadas a módulos. Foi necessário criar uma nova interface que suportasse essa funcionalidade. A maior dificuldade foi perceber como era suportado o modelo de permissões nas tabelas de metadados.

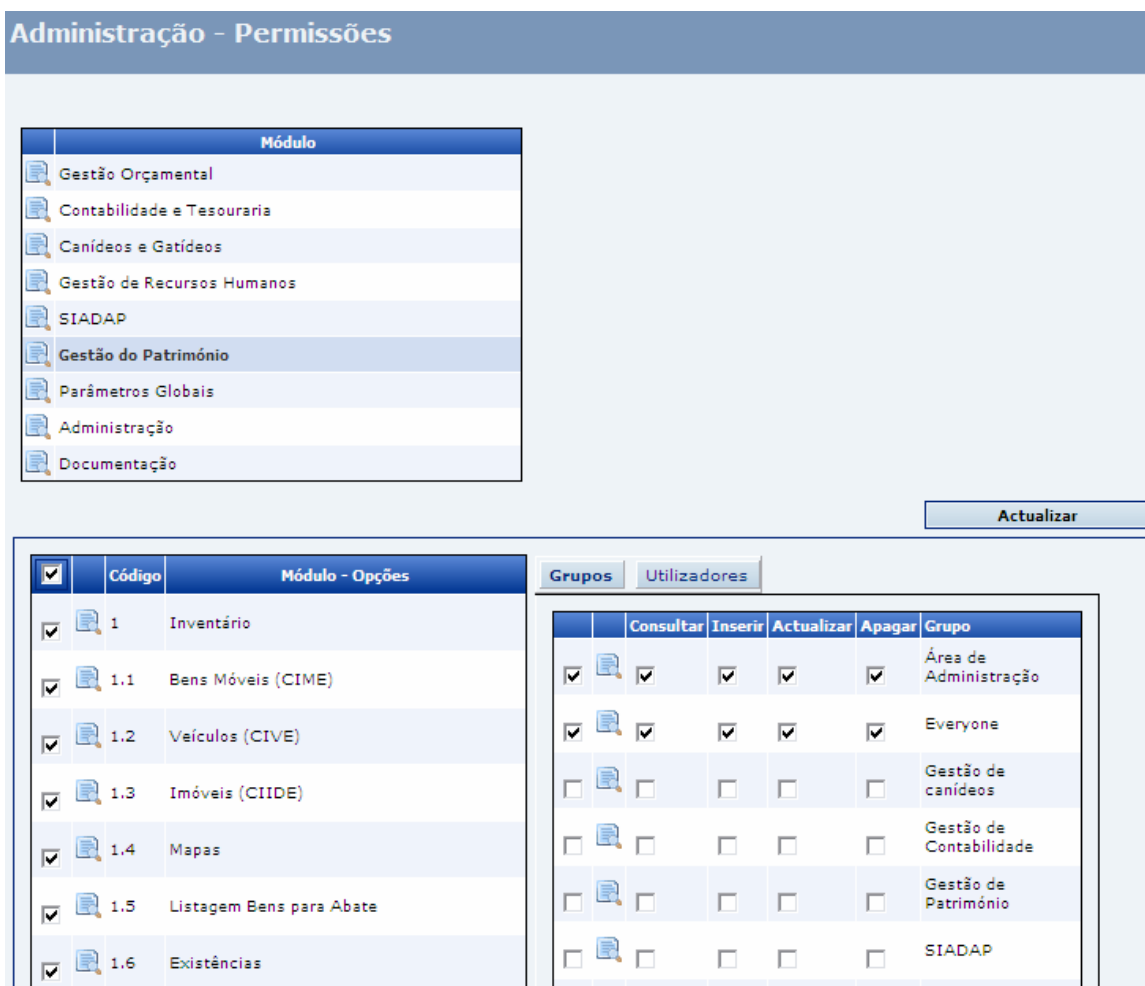


Fig. 5 - Administração - permissões

- **Implementação de um módulo de BackOffice**

O carregamento do menu e associação das permissões, páginas e tabelas, antes da implementação deste módulo era feito de forma muito rudimentar recorrendo a *scripts*. Para facilitar a inserção de registos nas tabelas de metadados foi necessário criar uma interface para uso interno, para a gestão da estrutura do menu e associação de permissões. Este módulo de *BackOffice* permite editar, inserir e remover registos nas tabelas de metadados, facilitando a alteração e inserção de registos nessas tabelas. Para implementação deste componente foi necessário perceber como se estruturavam os metadados.

Módulos

	CODmdAppModule	mdAppModule	Caption0	Order	Visible
Editar	GO	Gestão Orçamental	Gestão Orçamental	10	sim
Editar	CT	Contabilidade e Tesouraria	Contabilidade e Tesouraria	20	sim
Editar	CAN	Canídeos e Gatídeos	Canídeos e Gatídeos	30	sim
Editar	RH	Gestão de Recursos Humanos	Gestão de Recursos Humanos	40	sim
Editar	SAD	SIADAP	SIADAP	50	sim
Editar	GP	Gestão do Património	Gestão do Património	60	sim
Editar	PG	Parâmetros Globais	Parâmetros Globais	70	sim
Editar	AD	Administração	Administração	80	sim
Editar	DOC	Documentacao	Documentação	90	sim

Opções do Módulo
Administração

	COD	mdAppModuleOption	Caption0	Order	Visible	WebImage	WebLink
Editar	1	Utilizadores	Utilizadores	10	sim		../admin/aspXUtilizadoresListView.aspx
Editar	2	Grupos	Grupos	20	sim		../admin/aspXGruposListView.aspx
Editar	3	Permissões	Permissões	30	sim		../Admin/aspXPermissoesFormView.aspx

MDData

	mdData
Editar	mdEntity
Editar	mdApp

mdAppModuleOptionPage

	weblink
Editar	admin/aspXUtilizadoresFormView.aspx
Editar	Admin/aspXUtilizadoresListView.aspx

Fig. 6 - BackOffice gestão de tabelas de metadados

- **Criação de módulo de importação com dados de Cadastro para o SIADAP**

O SIADAP é uma aplicação para a Administração Pública em geral, e possibilita de uma forma eficaz, a gestão de todo o processo de avaliação de desempenho, segundo os termos legais.

Visto ser uma ferramenta que necessita à priori de ter os cadastros dos funcionários carregados para funcionar correctamente, foi necessário criar uma componente de importação destes dados a partir de outras aplicações / bases de dados. Caso contrário seria necessário carregar manualmente um a um, o que representaria demasiado tempo gasto. O meu trabalho consistiu em encontrar a relação entre os dois modelos de dados, e criar uma rotina que percorresse todos os registos da tabela de cadastro remota e carregasse a nossa local.

Após a importação de dados é exibido um relatório das operações efectuadas onde se assinalam eventuais erros ou inconsistências de dados. Esta funcionalidade é produzida à medida para cada cliente. Este trabalho, apesar de parecer fácil, é complexo pois é necessário encontrar um termo de comparação entre as tabelas. No final é necessário garantir que os dados fiquem coerentes na nossa aplicação.



Fig. 7 - Componente de importação

4.4. Funcionalidades implementadas

Esta aplicação integra no âmbito do POCAL e tem como função realizar de uma forma simples e rápida a gestão de inventário de um instituição. Este módulo permite efectuar toda a sua gestão desde a aquisição do Bem até ao seu abate. Permite também saber qual o valor real do bem num dado momento e suas variações patrimoniais ao longo do tempo. A imagem que se segue ilustra a interface inicial da aplicação.



Fig. 8 - Interface inicial

O menu apresenta-se sempre do lado esquerdo do ecrã. Os bens podem ser classificados por três tipos de categoria (CIME, CIIDE ou CIVE). Foi decidido tratar cada tipo de uma forma diferente visto cada um tinha atributos diferentes. Para aceder a cada um destes contextos basta seleccionar a opção desejada.

Todos os utilizadores desta aplicação devem possuir um nome de utilizador e uma senha própria. A figura seguinte ilustra a autenticação do utilizador, onde é requerido um *username* e *password*.

Fig. 9 - Autenticação do utilizador

Após a correcta autenticação, o utilizador tem acesso à aplicação e visualiza a interface descrita na figura 6. Esta aplicação POCAL servirá centenas de utilizadores, pelo que é de extrema importância que sejam definidos permissões para cada grupo de utilizadores. Numa instituição com muitos utilizadores é necessário que cada um realize o seu trabalho sem que comprometa o trabalho dos outros.

Importa referir que todos os contextos apresentam a sua informação de uma forma similar, no intuito de facilitar a sua utilização. Os contextos desta aplicação obedecem a um conceito genérico de interface, para que o utilizador analise a informação existente de uma forma simples e rápida.

Assim, na figura seguinte, é possível analisar a estrutura base de um contexto, e surge dividida em três partes consideradas fundamentais.

Descrição	Código Etiqueta	Data Inventário	Data Aquisição	abatido?	Relevância Cultural?
Mesa de reuniões oval	1030110-1		03-01-2006		
Secretária	1030112-4	20-01-2005	16-01-2005		
Telefone	1180202-7		11-09-2007		

Fig. 10 - Contexto genérico

1 – Área de pesquisa – neste contexto é possível pesquisar conteúdo existente através do auxílio de critérios ou filtros definidos pelo utilizador.

2 – Área de Inserção – nesta área é possível através de respectivos botões aceder à área de introdução de novos itens

3- Área de edição: Através do botão  é possível editar um registo.

No canto superior direito é possível alterar a *password*, ver o nome do utilizador que está ligado e enviar um e-mail com dúvidas e soluções que possam melhorar a aplicação. Esta opção é de extrema importância pois assim é possível receber opiniões do cliente, o que permite que a aplicação possa melhorar continuamente. Estas opções estão sempre visíveis.

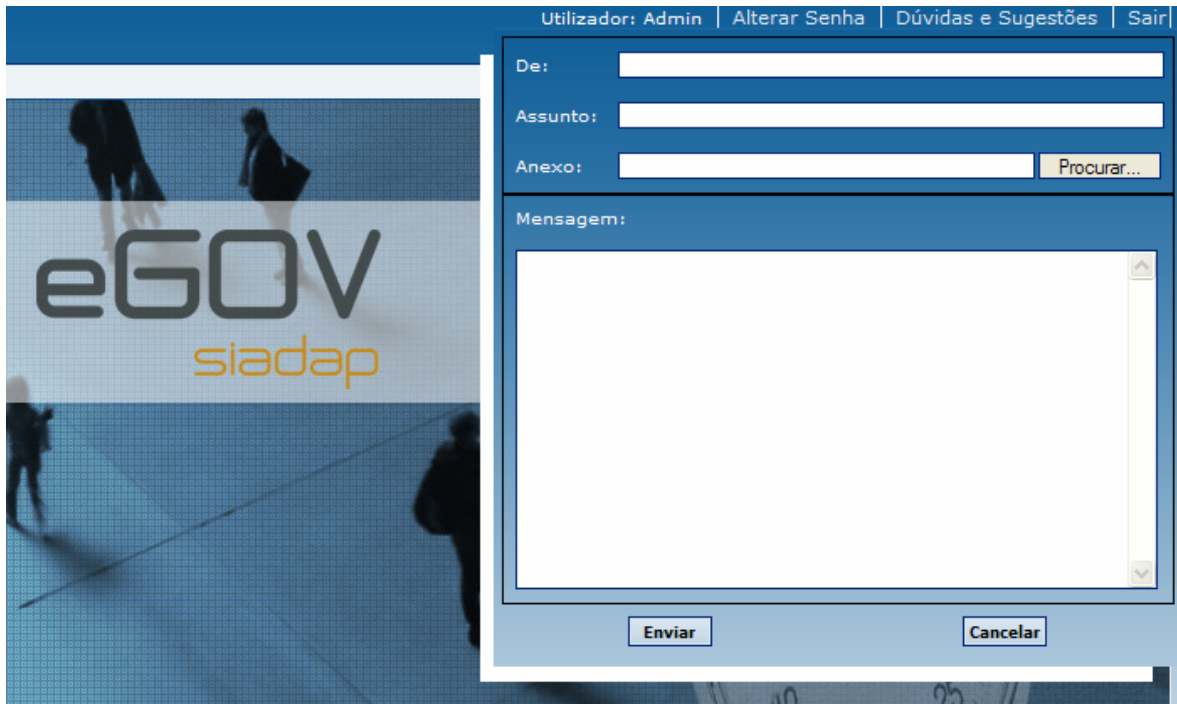


Fig. 11 - Módulo de Gestão de Património - feedback

Para criar um novo Bem, neste caso um Bem móvel, basta clicar no botão e inserir novo Bem. Aqui também se pode visualizar todos os Bens móveis existentes e realizar pesquisas sobre os mesmos.



Fig. 12 - Listagem dos bens móveis (CIME)

A identificação de cada bem é composta pelo código do classificador geral e número de inventário, posteriormente é possível imprimir etiquetas com sua identificação. A aplicação está preparada para imprimir etiquetas que podem ser depois coladas nos Bens.

No exemplo a seguir foi seleccionado dois Bens, e ao carregarmos no botão “imprimir etiquetas” é gerado dinamicamente uma folha com as etiquetas que pretendemos imprimir.



Fig. 13 - Geração de etiquetas para identificação dos Bens

Ao seleccionarmos um Bem podemos visualizar e alterar toda a informação referente a esse Bem, desde a sua classificação, o valor inicial, vida útil estimada, código de identificação, variações patrimoniais, etc ...

Requisição de Bem		
Requisição Oficial	Livro Req.Oficial	Dt.Req.Oficial

Entidade Fornecedora		
Entidade	Nº da Factura	Dt.Factura
IKEA	2039/06	03-02-2006

Custos de Aquisição		
Tipo do Valor Inicial	Valor S/IVA	Tx. IVA
-- Inserir Tipo de Valor Inicial --	250,00	21% - IVA Tx.Normal - Continente
Despesas de Compra	Valor Inicial (Total)	
0,00	302,50	

Abate	
<input checked="" type="checkbox"/> A Abater	Tipo de Abate
	01 - Alineação a título oneroso

Fig. 14 - Ficha do Bem para consulta ou edição

Todas as variações patrimoniais (quer de valor, quer de tempo de vida útil) de um item devem ser inseridas no tabulador Variação Patrimonial. Estas variações podem incluir valorizações em que o Bem mantém ou aumenta o seu próprio valor (ou desvalorizações, onde o Bem perde valor). Exemplo de valorizações comuns são reparações. Exemplos de desvalorizações são sinistros, degradação excepcional do material. É através desta lista que as amortizações a serem calculadas posteriormente podem sofrer alterações, pelo que é de extrema importância manter esta lista sempre actualizada.

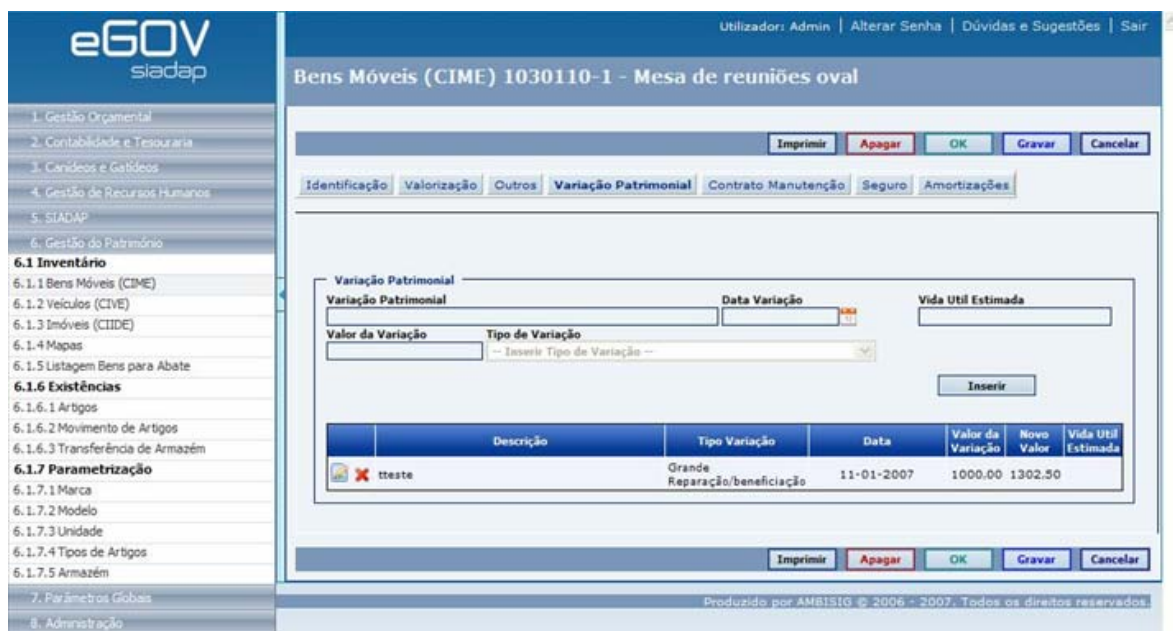


Fig. 15 - Variações patrimoniais de um Bem

O tabulador de Amortizações possibilita saber o valor real do Bem num determinado instante. Este cálculo é feito com base no valor inicial, o tempo de vida do Bem e suas variações patrimoniais. Esta listagem também pode ser gerada no formato PDF para impressão

Imprimir Apagar OK Gravar Cancelar

Identificação Valorização Outros **Variação Patrimonial** Contrato Manutenção Seguro **Amortizações**

Calcular Imprimir

Número	Restante	Ano	Mês	Taxa Mês	Valor Mês	Taxa Ano	Acumulado Ano	Total Acumulado	Total Líquido
1	96	2006	2	1,04	3,15	2,08	3,15	3,15	299,35
2	95	2006	3	1,04	3,15	3,12	6,30	6,30	296,20
3	94	2006	4	1,04	3,15	4,16	9,45	9,45	293,05
4	93	2006	5	1,04	3,15	5,20	12,60	12,60	289,90
5	92	2006	6	1,04	3,15	6,24	15,75	15,75	286,75
6	91	2006	7	1,04	3,15	7,28	18,90	18,90	283,60
7	90	2006	8	1,04	3,15	8,32	22,05	22,05	280,45
8	89	2006	9	1,04	3,15	9,36	25,20	25,20	277,30
9	88	2006	10	1,04	3,15	10,40	28,35	28,35	274,15
10	87	2006	11	1,04	3,15	11,44	31,50	31,50	271,00
11	86	2006	12	1,04	3,15	12,48	34,65	34,65	267,85
12	85	2007	1	1,04	3,15	1,04	3,15	37,80	264,70
13	84	2007	2	1,19	15,06	2,38	18,21	52,86	1249,64
14	83	2007	3	1,19	15,06	3,57	33,27	67,92	1234,58
15	82	2007	4	1,19	15,06	4,76	48,33	82,98	1219,52
16	81	2007	5	1,19	15,06	5,95	63,39	98,04	1204,46
17	80	2007	6	1,19	15,06	7,14	78,45	113,10	1189,40

Fig. 16 - Listagens das amortizações

Depois de preenchida a informação relativa a um Bem, é possível imprimir a sua ficha, no formato definido no decreto lei. Para a produção deste relatório foi usada a aplicação *Crystal Report*.

Ficha de Identificação de Bens Móveis

F1 - CIME

Entidade de Demonstração
morada da entidade
0000-000 localidade
Concelho: concelho
Contribuinte nº: 100000001



1. IDENTIFICAÇÃO DO BEM

Nº de inventário: 1 Designação :Mesa de reuniões oval

Classificação do Bem: Classe: 103 - Equipamento e material de escritório e de reprografia Tipo: 10301 - Mobiliário Bem: 1030110 - Mesas		Localização: Edifício: Sede Piso: 0 Sala: 1	
Tipo de Aquisição: Aquisição por compra	Data de Aquisição: 03-01-2006	Entidade Fornecedora: IKEA	Nº Fatura: 2039/06
Requisição Oficial:	Livro	Data	Data de Ent. em funo: 03-01-2006
			Vida Estimada: 8
			Tx. Amortização: 12,50
			Amortização Anual: 37,81

2. DESCRIÇÃO

Ref 1 Marcos: Desconhecido ou N/A	Ref 2 Modelo: Desconhecido ou N/A	Ref 3 Medidas: (Centímetros)	Comprimento 210	Largura 120	Altura 70
--------------------------------------	--------------------------------------	---------------------------------	--------------------	----------------	--------------

3. VALORIZAÇÃO E REGISTO

Custos:	Aquisição	Produção	Avaliação	Despesas de Compra 0,00	Custo Total 302,50		
Contrato de Locação:	Valor Total 0,00	Nº Rendas 0	Valor/Rendas 0,00	Início	Termo	Valor Residual 0,00	Opção/Compra Não
Classificação da Despesa:	Económios			Orçamental			
	426 - Equipamento administrativo						

4. ABATE

Tipo de Abate: Alineação a título oneroso	Autorização:	Amortiz. Acumulada:	Receita Cobrada:	Classificação da Receita:	Data Lançamento:
--	--------------	---------------------	------------------	---------------------------	------------------

5. CONTROLO PATRIMONIAL

Inventário:	Data do Últ. Controlo
Estado de Conservação:	
Operacionalidade:	

6. AFECTAÇÃO

Funcionário:
Departamento:

7. VARIAÇÕES PATRIMONIAIS

Grande Reparação/beneficiação	Tipo	Valor 1.000,00	Vida útil não alterada	Mês/Ano 01/2007	Valor 1.302,50
-------------------------------	------	-------------------	---------------------------	--------------------	-------------------

8. SEGUROS

Fig. 17 - Ficha de identificação dos Bens móveis

Posteriormente é possível a impressão de mapas de dados de acordo com dados carregados, para isso é necessário seleccionar no menu a opção Mapas

Mapas

Mapas

Mapa Síntese 1989

de
todos os valores

até
todos os valores

Fig. 18 - Contexto mapas

Dada a complexidade dos mapas gerados foi necessário recorrer a tabelas temporárias para a realização dos cálculos e com um *timestamp* como identificador, depois de gerados os relatórios desses registos são removidos da base de dados.

Em anexo encontram-se dois mapas gerados pela aplicação (Anexo 1 e 2).

5. Conclusão

O ambiente no mundo real do desenvolvimento de *software* é bastante diferente daquele encontrado no mundo académico. O ritmo é bastante mais acelerado e as pressões bastantes maiores, mas também o trabalho realizado é mais significativo e gratificante. A aprendizagem de novas tecnologias e o contacto com tecnologias ainda em desenvolvimento é muito cativante.

Em termos de experiência profissional, foi muito gratificante pôr em prática os conhecimentos académicos. Apesar de, ao longo do curso não ter aprendido a trabalhar especificamente com o ASP.NET, não foi um problema, pois, uma das principais coisas que desenvolvemos na faculdade é a capacidade de nos adaptarmos e ultrapassarmos os problemas que vão aparecendo.

A integração na empresa e na equipa correu bem, fui inserido numa equipa onde existiu uma enorme entre-ajuda. Foi importante trabalhar com pessoas mais experientes, e durante todo o estágio existiu muita troca de conhecimentos, o que foi importante para uma rápida adaptação.

Fazendo uma retrospectiva de todo o trabalho desenvolvido e de todos os conhecimentos adquiridos, estou convicto que cumpri os objectivos que me foram propostos pela da empresa e as expectativas e responsabilidades por mim depositados, bem como os objectivos do estágio.

6. Bibliografia

Ref. 1, Manual da Qualidade da AmbiSIG, 2007

Ref. 2, Manual do Utilizador da Gestão de Património, 2007

Ref. 3, Diário da República – legislação sobre o Cadastro e Inventário dos Bens do Estado

Ref. 4, POCAL comentado – Plano Oficial de Contabilidade das Autarquias Locais, ISBN 972-51-1087-0

Ref. 5, ASP.NET Curso Completo, Luís Abreu

Ref. 6, Crystal Reports 9 - Guia Prático, Carlos Montoya, Fábio Rafael Ciocari

Ref. 7, JavaScript a Bíblia, Danny Goodman

Ref. 8, RYourdon, Edward. Análise Estruturada Moderna. Ed. Campus, 1995

Ref. 9, Avison & Fitzgerald 1995, Information Systems Development: Methodologies, Techniques and Tools 2nd ed. McGraw Hill, Maidenhead

7. Referências WWW

- www.dgaa.pt site oficial da Direcção Geral das Autarquias Locais
- www.asp.net site oficial de ASP.NET
- www.ambisig.pt site da empresa
- www.crystal-reports-software.com software Crystal Reports

8. Lista de Siglas e Acrónimos

VSS	Visual Source Safe
POCAL	Plano Oficial de Contabilidade das Autarquias Locais
CIBE	Cadastro de Inventário de Bens de Estado
CIME	Cadastro de Inventário dos Móveis do Estado
CIVE	Cadastro de Inventário dos Veículos do Estado
CIIDE	Cadastro de Inventário dos Móveis do Estado
WEB	Rede de Alcance Mundial
BackOffice	Administração
Norma EN ISO 9001	Especificação de requisitos para um sistema de gestão da qualidade
Intranet	Rede Interna
ASP.NET	Plataforma da Microsoft para desenvolvimento de aplicações WEB
PDF	Portable Document Format
IIS	Internet Information Services
VB6	Visual Basic 6
Client-tier	Camada de Apresentação
Middle-tier	Camada lógica da Aplicação
Database-tier	Base de dados
SQL	Structured Query Language
BD	Base de dados
JS	JavaScript
C# e VB.NET	Linguagens de Programação
IP	Internet Protocol
XMLHTTP	Extensible Markup Language HyperText Transfer Protocol
AJAX	Asynchronous JavaScript and XML
Atlas	Ferramenta Microsoft que junta ASP.NET com AJAX
DHTML	Dynamic HyperText Markup Language
HTML	HyperText Markup Language

9. Anexos

Ficha de Amortizações

Ano:2008



CIBE
 Cadastro e Inventário dos Bens de Estado
 Entidade para Demonstração
 Av. Infante Santo, 69-H
 1250-900 Lisboa
 Lisboa
 202202101

Identificação dos Bens		Número de Inventário	Descrição	Data do fecho patrimonial mês/ano	Tipo de Aquisição	Valores de aquisição, produção ou outro valor contabilizado	Variação Patrimonial		Valor Patrimonial Atualizado	Número de anos de vida esperada	Amortizações			Valor Patrimonial Líquido			
							Tipo de variação	Valor			de Anos Anteriores	Do ano	A acumuladas				
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16-18+15	19-11-16	
103	01	10	1	Mesa de reuniões oval	1/2006	01	1.302,50		0,00	1.302,50	6	203,46	14,28	180,72	384,18	918,32	
103	01	12	4	Secretaria	1/2005	01	120,00		0,00	120,00	5	43,75	12,48	15,00	58,75	61,25	
202	04	03	5	Viatura para transporte de idosos	1/2006	01	1.000,00		0,00	1.000,00	3	383,41	20,04	200,04	583,45	416,55	
301	01	03	6	Oficina e Armazem	1/2003	01	117.500,00		0,00	117.500,00	11	11.578,96	8,28	9.556,80	21.135,76	96.364,24	
401	02	01	3	Edifício Sede	1/1989	05	372.500,00		0,00	372.500,00	1	336.475,52	50,04	33.249,50	369.725,12	2.770,88	
Total Geral ou a Transportar																	
										492.422,50							100.531,24

Ficha de Inventário

Ano:2008



CIBE
 Cadastro e Inventário
 dos
 Bens de Estado
 Entidade para Demonstração
 Av. Infante Santo, 68-H
 1250-990 Lisboa
 Lisboa
 202202101

Índice	Identificação dos Bens				Ajustamento do valor			Alterações Patrimoniais			Vida Útil Esperada	Valor Patrimonial Atualizado	Abate			
	Tipo de Bem	Bem	Numero de Inventário	Descrição	Tipo de Aquisição	Ano	Valor	Tipo de Alterações	Ano	Valor			Tipo de Abate	Ano	Rubrica Orçamental	Valor
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
103	01	10	1	Mesa de reuniões oval	01	2006	302,50	GR	2007	1.000,00	6	1302,50				0,00
103	01	12	4	Secretaria	01	2005	130,00			0,00	5	130,00				0,00
116	01	08	7	teste	03	2007	1.000,00	DE	2007	-100,00	0	900,00	02	2007	distof	100,00
202	01	03	2	Viatura para transporte escolar	01	2001	30.345,00			0,00	0	30345,00	01	2007	Gr. Carlos	1.000,00
202	04	03	5	Viatura para transporte de idosos	01	2006	1.000,00			0,00	3	1000,00				0,00
301	01	03	6	Oficina e Armazem	01	2003	105.000,00	GR	2007	12.500,00	11	117500,00				0,00
401	02	01	3	Edificio Sede	05	1989	340.000,00	GR	2007	32.500,00	1	372500,00				0,00
Total Geral ou a Transportar												477.767,50	45.900,00	523.667,50	1.100,00	

