



Universidade do Porto  
Faculdade de Engenharia

**FEUP**



Miguel de Castro Pereira

## Portal de Serviços na Vodafone Telecel, SA

LEIC  
PERm

**LEIC**  
Porto, 2002



# **Portal de Serviços na Vodafone Telecel S.A.**

*Miguel de Castro Pereira*

**Relatório do Estágio Curricular da LEIC 2001/02**

Orientador na FEUP: Prof. Manuel Ricardo

Orientador na Vodafone Telecel S.A: Eng. Henrique Necho



**Faculdade de Engenharia da Universidade do Porto  
Licenciatura em Engenharia Informática e Computação**

2000-09-16

**UNIVERSIDADE DO PORTO**  
Faculdade de Engenharia  
**BIBLIOTECA** H  
N.º 65318  
CDU 004 (047.3)  
Data 14 / 3 / 20 03

## Resumo

Neste documento é apresentado o projecto “Portal de Serviços” desenvolvido na empresa Vodafone Telecel S.A , que se centra na criação de um portal seguro utilizando a plataforma de construção de portais empresariais SunONE *iPlanet Portal Server 3.0*.

Este projecto surge com a necessidade de testar soluções que, com segurança, facilitem a introdução de novos serviços no âmbito da rede de distribuição da Vodafone Telecel S.A. São apresentadas as tecnologias e produtos utilizados e a forma como se integraram aplicações internas já existentes nesta plataforma e as soluções encontradas são descritas com detalhe.

A solução para a validação de uma máquina cliente no “Portal de Serviços” foi uma ramificação do projecto inicial e é descrita com pormenor, tanto a nível do seu desenho como no âmbito da sua adopção por uma aplicação interna da empresa. Em anexo encontram-se exemplos de actas de reuniões, detalhes de implementação do projecto e planeamento interno.

Este trabalho permitiu avaliar o produto iPlanet Portal Server 3.0 e as mais-valias que do seu uso podem decorrer. Além disto, a solução encontrada para a identificação única de um cliente que acesse a um sítio Web foi aproveitada numa aplicação interna da Vodafone Telecel S.A., estando já implementada em fase de produção.



## **Agradecimentos**

Em primeiro lugar ao Joel Alvim, meu companheiro de projecto e de estágio.

Às pessoas que bem me receberam na Vodafone Telecel S.A, nomeadamente ao Eng. Rui Carvalho e ao Eng. Jorge Sousa, que me proporcionaram um ambiente de trabalho agradável e produtivo.

Ao meu orientador de estágio da FEUP, Prof. Manuel Ricardo, ao meu orientador de estágio na Vodafone Telecel S.A, Eng. Henrique Necho e ao director da cadeira Prof. Raul Vidal.

Por último a todos os meus amigos e pessoas que apesar de não directamente envolvidas me ajudaram nos momentos difíceis e estiveram presentes nos momentos agradáveis!

Índice de Conteúdos

- 1 Introdução..... 1
  - 1.1 Sobre este Relatório de Estágio .....1
  - 1.2 Apresentação da Vodafone Telecel S.A. ....1
    - Visão e Valores Fundamentais .....1
    - Perfil da Empresa .....2
    - Estratégia e Diferenciação.....2
    - Breve História .....3
    - Normas Organizacionais.....3
  - 1.3 O Projecto “Portal de Serviços” .....4
    - Enquadramento do Projecto .....4
    - Aplicações Internas Envolvidas .....5
    - Ramificações do Projecto “Portal de Serviços” .....5
- 2 Tecnologias e Produtos Utilizados ..... 6
  - 2.1 Tecnologias.....6
  - 2.2 Software Utilizado .....11
- 3 Portal de Serviços ..... 18
  - 3.1 Sítio Web Interno do Projecto .....18
  - 3.2 Portal de Serviços.....18
    - Introdução .....18
    - Arquitectura Física do Projecto Piloto.....18
    - Configurações do iPlanet Portal Server.....20
    - Implementação .....21
  - 3.3 Aplicação Existente: Web Parceiros .....25
    - Introdução .....25
    - Implementação do *Single Sign On* .....25
  - 3.4 Aplicação Existente: SFAWeb .....26
    - Introdução .....26
    - Implementação do *Single Sign On* .....26
  - 3.5 Aplicação Nova: Feedback Reports.....27
  - 3.6 Replicação de Utilizadores.....28
    - Introdução .....28
    - Requisitos Iniciais .....28
    - Arquitectura.....28
    - Descrição da Implementação .....29
- 4 Ramificações do Projecto Portal de Serviços ..... 34
  - 4.1 Validação do *hardware* cliente .....34
    - Requisitos Iniciais .....34
    - Tecnologias Relevantes.....34
    - Arquitectura.....37
    - Revisão aos Requisitos Iniciais .....37
    - Descrição da Implementação .....38
  - 4.2 Validação do *hardware* cliente na aplicação WebParceiros.....44
    - Requisitos Iniciais .....44

|                                                                               |    |
|-------------------------------------------------------------------------------|----|
| Análise .....                                                                 | 46 |
| Descrição da Implementação .....                                              | 50 |
| 5 Conclusões .....                                                            | 52 |
| 5.1 Considerações Pessoais.....                                               | 52 |
| 5.2 Contribuições .....                                                       | 52 |
| 5.3 Trabalho Futuro .....                                                     | 52 |
| Referências e Bibliografia.....                                               | 53 |
| ANEXO A – Detalhes de Implementação .....                                     | 54 |
| A.1 Aplicação Feedback Reports .....                                          | 54 |
| Motivação e Requisitos .....                                                  | 54 |
| Arquitectura.....                                                             | 54 |
| Análise .....                                                                 | 55 |
| Descrição da Implementação .....                                              | 56 |
| A.2 Protótipo Gráfico para a integração de novas opções no WebParceiros ..... | 58 |
| Opções Master User .....                                                      | 58 |
| A.3 Sítio Web Interno de Projecto .....                                       | 63 |
| Requisitos e Análise .....                                                    | 63 |
| Descrição da Implementação .....                                              | 64 |
| ANEXO B – Planeamento do estágio e comentário.....                            | 67 |
| Planeamento do Estágio .....                                                  | 67 |
| Fases e Momentos Importantes do Estágio .....                                 | 68 |
| ANEXO C – Testes Unitários.....                                               | 71 |
| Teste de Registo de Equipamento .....                                         | 71 |



## Índice de Figuras

|                                                                                                 |    |
|-------------------------------------------------------------------------------------------------|----|
| Fig. 2-1 Tecnologias - Estado actual da gama de plataformas JAVA .....                          | 6  |
| Fig. 2-2 Tecnologias - Arquitectura típica J2EE .....                                           | 7  |
| Fig. 2-3 Tecnologias - Exemplo de directório LDAP .....                                         | 9  |
| Fig. 2-4 Tecnologias - Entrada LDAP com múltiplos atributos .....                               | 10 |
| Fig. 2-5 Software Utilizado - Arquitectura do iPlanet Portal Server .....                       | 12 |
| Fig. 2-6 Software Utilizado - Exemplo de árvore de papéis baseada em LDAP .....                 | 13 |
| Fig. 2-7 Software Utilizado - Arquitectura da Netlet .....                                      | 14 |
| Fig. 2-8 Software Utilizado - Camadas do BEA Weblogic 5.1 .....                                 | 15 |
| Fig. 3-1 Portal – Arquitectura do projecto piloto .....                                         | 19 |
| Fig. 3-2 Portal - Administração .....                                                           | 20 |
| Fig. 3-3 Portal - Dependências entre Entidades do iPlanet Portal Server .....                   | 21 |
| Fig. 3-4 Portal - Estrutura LDAP criada .....                                                   | 22 |
| Fig. 3-5 Portal – Novo componente desenvolvido iwtUserMacs .....                                | 23 |
| Fig. 3-6 Portal – Novo fluxo de <i>login</i> .....                                              | 23 |
| Fig. 3-7 Portal – Mudança na interface gráfica .....                                            | 24 |
| Fig. 3-8 Portal – Área de Trabalho .....                                                        | 24 |
| Fig. 3-9 WebParceiros – Fluxo de <i>login</i> antes de Single Sign On .....                     | 25 |
| Fig. 3-10 WebParceiros – Fluxo de <i>login</i> com Single Sign On implementado .....            | 26 |
| Fig. 3-11 Replicação - Arquitectura Lógica .....                                                | 29 |
| Fig. 3-12 Replicação - Diagrama de estados .....                                                | 31 |
| Fig. 4-1 Validação de Hardware - Arquitectura SNMP .....                                        | 36 |
| Fig. 4-2 Validação de Hardware - Diagrama de Componentes .....                                  | 37 |
| Fig. 4-3 Validação de Hardware - Arquitectura Implementada .....                                | 39 |
| Fig. 4-4 Validação de Hardware - Arquitectura da biblioteca - SNMP .....                        | 40 |
| Fig. 4-5 Validação de Hardware - Diagrama de Estados da Applet de Instalação .....              | 41 |
| Fig. 4-6 Validação de Hardware - Aviso do Internet Explorer .....                               | 44 |
| Fig. 4-7 Validação de Hardware no WebParceiros - Casos de Uso .....                             | 46 |
| Fig. 4-8 Validação de Hardware no WebParceiros - Diagrama de Sequência do <i>Login</i> .....    | 47 |
| Fig. 4-9 Validação de Hardware no WebParceiros - Diagrama de Sequência da Activação .....       | 48 |
| Fig. 4-10 Validação de Hardware no WebParceiros - Fluxo de acesso de um utilizador NORMAL ..... | 49 |
| Fig. 4-11 Validação de Hardware no WebParceiros – Novo modelo de dados .....                    | 50 |
| Fig. A-1 Feedback Reports - Arquitectura .....                                                  | 54 |
| Fig. A-2 Feedback Reports - Diagrama de Classes .....                                           | 55 |

Fig. A-3 Feedback Reports Diagrama de Casos de Uso .....56

Fig. A-4 Feedback Reports - Página de Inserção de Feedback .....57

Fig. A-5 Feedback Reports Página de Visualização de Estatísticas.....57

Fig. A-6 Novas Opções WebParceiros - Menu com novo grupo de opções .....58

Fig. A-7 Novas Opções WebParceiros - Passo “Escolher” do registo .....59

Fig. A-8 Novas Opções WebParceiros - Passo “Confirmar” do registo .....59

Fig. A-9 Novas Opções WebParceiros - Passo de “Resultado” de um registo .....60

Fig. A-10 Novas Opções WebParceiros - Passo “Escolher” da gestão de equipamento .....60

Fig. A-11 Novas Opções WebParceiros - Operação de listagem da gestão de equipamento.....61

Fig. A-12 Novas Opções WebParceiros - Operação de Edição da gestão de equipamentos .....61

Fig. A-13 Novas Opções WebParceiros - Confirmação da mudança de nome do equipamento .....62

Fig. A-14 Novas Opções WebParceiros - Confirmação da desactivação .....62

Fig. A-15 Novas Opções WebParceiros - Resultado da Operação .....62

Fig. A-16 Sítio Web Interno de Projecto - Diagrama de Casos de Uso .....63

Fig. A-17 Sítio Web Interno de Projecto - Interface.....64

Fig. A-18 Sítio Web Interno de Projecto – Inserção de Documento.....64

Fig. A-19 Sítio Web Interno de Projecto – Inserção de Novo Grupo.....65

Fig. A-20 Sítio Web Interno de Projecto – Listagem da Documentação .....65

Fig. A-21 Sítio Web Interno de Projecto – Diagrama de Classes .....66

# 1 Introdução

## 1.1 Sobre este Relatório de Estágio

O projecto “Portal de Serviços” foi desenvolvido por mim e pelo meu colega de estágio Joel Alvim em estreita colaboração. Assim, como combinado com os orientadores de estágio, o capítulo 2 foi escrito em co-autoria.

Para além da co-autoria, existe uma ressalva a ser feita: nesta empresa fomos sempre tratados como uma equipa de projecto e não houve uma separação formal do trabalho desde o início. Assim, existem assuntos neste relatório que são resumos do que “foi feito” sem explicações de como foi feito, querendo isto dizer que foi o colega de estágio que se centrou mais nessa parte. Contudo, com o desenrolar do projecto, acabamos naturalmente por seguir alguns caminhos diferentes e a separação prática acabou por ser conseguida.

Este relatório está organizado segundo uma estrutura simples e lógica. Os capítulos do corpo do relatório são “orientados ao projecto”, ou seja, descrevem vários projectos e respectivos sub-projectos. Cada um destes componentes é normalmente organizado segundo uma estrutura típica (p.ex., Introdução, Requisitos, Análise, Implementação).

No capítulo 1 é feita uma breve introdução do projecto, da empresa e das suas normas internas. No capítulo 2 temos a descrição das tecnologias e produtos transversais ao projecto. No capítulo 3 é feita a descrição do projecto “Portal de Serviços” e de todos os componentes desenvolvidos no âmbito da implementação da solução, incluindo novas aplicações, integração de aplicações existentes e replicação de utilizadores. O capítulo 4 divide-se em duas partes: a primeira, “Validação do *Hardware* Cliente”, descreve detalhadamente esta solução que foi originalmente desenvolvida para integrar no “Portal de Serviços” mas que foi implementada no âmbito de uma ramificação do projecto, descrita na segunda parte do capítulo. As conclusões e comentários finais são apresentados no capítulo 5. No anexo A temos detalhes de implementação pertencentes a várias partes do relatório. No anexo B existe o planeamento inicial do projecto e o que foi realmente feito, juntamente com descrições e marcos importantes. Finalmente, o anexo C contém alguns exemplos de Teste Unitários realizados no âmbito da implementação da Validação do *hardware* cliente numa aplicação interna Vodafone Telecel S.A.

## 1.2 Apresentação da Vodafone Telecel S.A.

A Vodafone Telecel S.A. Telecel é uma empresa de telecomunicações e serviços de voz e dados que resulta da introdução da Telecel S.A. no grupo Vodafone, sediado no Reino Unido, em Janeiro de 2001. Possui instalações um pouco por todo o país desde lojas e agentes a centros de dados e de redes

### Visão e Valores Fundamentais

#### A Visão do Grupo Vodafone

Ser o líder mundial em comunicações móveis - enriquecendo a vida dos clientes, ajudando a estabelecer ligações entre as pessoas, as empresas e as comunidades, num mundo em movimento.

## **A Visão Portuguesa**

Ser reconhecidos como a melhor empresa portuguesa, líder no desenvolvimento das comunicações móveis, contribuindo assim para o bem-estar das pessoas, das empresas e da sociedade em geral.

## **Valores corporativos**

- Paixão pelos clientes
- Paixão pelas nossas pessoas
- Paixão pelos resultados
- Paixão pelo mundo que nos rodeia

## **Perfil da Empresa**

A Vodafone Telecel S.A em Portugal é um operador de telecomunicações que possui desde 1992 uma rede móvel GSM (Global System for Mobile Communications) sendo esse neste momento um dos seus campos de acção. O seu objectivo é criar ofertas de serviços com qualidade e valor reconhecidamente superiores de modo a satisfazer as necessidades de comunicação dos clientes.

O seu posicionamento actual é definido como um operador de “Telecomunicações e Informação”, com um negócio base nas comunicações móveis e uma aposta na oferta de serviços móveis de comunicação de dados. Um dos seus objectivos mais importantes é constituir um referencial de gestão eficiente e profissional, prestando um serviço ao público de elevada qualidade.

De modo a manter uma posição de liderança nos segmentos empresariais do mercado celular, a Vodafone Telecel S.A desenvolve ofertas complementares de serviços fixos e participa em projectos de Tecnologia de Informação e Redes que contribuem para o aumento das receitas de comunicação e para a sempre importante fidelização dos clientes.

## **Estratégia e Diferenciação**

A estratégia da Vodafone Telecel S.A assenta na disponibilização de uma oferta muito competitiva em todas as vertentes da sua actividade, de modo a cativar e manter os clientes mais valiosos de cada segmento do mercado.

O principais factores de diferenciação da Vodafone Telecel S.A são:

- Uma rede de comunicações celulares alargada e de elevada qualidade
- Um Serviço de Apoio a Clientes de excelência
- Liderança e inovação em *marketing*
- Oferta diversificada e competitiva
- Múltiplos canais de distribuição profissionais e dinâmicos

A estratégia futura da Vodafone Telecel S.A consiste em continuar a ser uma empresa em crescimento, consolidando e reforçando a sua quota de mercado nos próximos 4 a 5 anos, assegurar a estrutura de custos e rentabilidade do investimento mais competitivas do mercado, ser a melhor marca de serviços móveis em termos do grau de reconhecimento e prestígio, ser



líder na inovação e novas aplicações e serviços e, finalmente, aproveitar ao máximo as sinergias com o grupo Vodafone S.A.

## **Breve História**

### ***Início de Actividade***

A Vodafone inaugurou a sua actividade comercial, como Telecel S.A, a 18 de Outubro de 1992, disponibilizando de imediato ao público um serviço de comunicações celulares GSM totalmente operacional e cobrindo 57% do território e 83% da população de Portugal.

A Telecel conseguiu ter a sua rede celular em funcionamento um ano após ter obtido a sua licença de exploração, batendo o recorde mundial da instalação mais rápida de uma rede GSM.

### ***Marca Vodafone***

Em Janeiro de 2001 a empresa passou a designar-se por Telecel Vodafone S.A. e foi iniciado um processo de migração que culminou com a alteração definitiva para Vodafone Telecel S.A em Outubro desse ano.

Este processo reflecte a ligação da empresa ao maior grupo mundial de comunicações móveis e decorreu em simultâneo com outros operadores europeus do grupo Vodafone; no entanto, a Telecel Vodafone foi a primeira empresa do grupo a reunir as condições necessárias para a mudança definitiva da marca, em Outubro de 2001.

O grupo Vodafone beneficia de uma imagem de marca global, reconhecida em todo o mundo e que presta serviços a cerca de 103,9 milhões de clientes em 28 países. A inserção num grupo destas características permite oferecer ao cliente preços cada vez mais competitivos e serviços e equipamentos cada vez melhores e mais diferenciados.

### ***Certificação Global de Qualidade***

Em Março de 2002 a empresa dá mais um passo na inovação em Portugal, tornando-se no primeiro operador de telecomunicações a obter certificação de qualidade NP EN ISO9001:2000 pela Associação Portuguesa de Certificação(APCER).

## **Normas Organizacionais**

A Vodafone, como qualquer grande empresa, dispõe de um conjunto variado de normas de estruturação de projectos que permitem uma maior organização global.

Assim, um projecto interno está sob a alçada de uma das direcções da empresa e passa por quatro fases fundamentais, a saber:

- Análise de Requisitos Gerais
- Desenvolvimento, fase em que se trabalha sobre um ambiente separado não necessariamente igual ao das fases posteriores e em que se vão introduzindo alterações à configuração e ao código. Os testes nesta fase são feitos pela própria equipa de desenvolvimento.
- Pré-Produção, ou *system test*, fase em que o projecto é passado para uma configuração igual ou próxima da configuração de produção e são feitos testes de funcionalidades e performance por parte de uma equipa especialmente formada para o efeito

- Produção, fase em que o projecto se encontra ao dispor dos utilizadores finais.

Estas três fases coexistem durante todo o ciclo de vida do projecto, respeitando-se sempre as regras de introdução de mudanças.

Os fluxos de documentação seguem também normas estritas e, entre outros documentos, são produzidos em fase de desenvolvimento os chamados *General Design*<sup>1</sup> e *Detail Design*<sup>2</sup>, documentos estes posteriormente entregues à equipa de Pré-Produção

## 1.3 O Projecto “Portal de Serviços”

### Enquadramento do Projecto

A Vodafone, como empresa competitiva que é, possui uma panóplia de aplicações actualmente em uso e que suportam todo o seu negócio, em áreas tão distintas como seja nas bases de dados, aplicações de gestão das relações com clientes (CRM, *Costumer Relationship Management*), servidores aplicativos que alojam toda a lógica de negócio inerente aos seus processos, contabilidade e suporte ao planeamento de recursos (ERP, *Enterprise Resource Planner*), gestão documental, workflow, correio electrónico corporativo, etc.

Este projecto destina-se a criar um portal empresarial que forneça os mais variados serviços a parceiros Vodafone, desde as aplicações supracitadas até novas funcionalidades. Este objectivo deve ser conseguido com o máximo de segurança possível e de maneira a que se torne fácil para um parceiro usar as aplicações, bastando para isso introduzir os dados de autenticação para entrar no portal e utilizar todas as funcionalidades ao seu dispor.

O projecto “Portal de Serviços” foi introduzido na secção de “Novos Projectos” e foi criada uma equipa de projecto com as seguintes características:

- Gestores: Eng. Henrique Necho e Eng. Rui Carvalho
- Equipa de desenvolvimento: Miguel Pereira e Joel Alvim

Numa fase posterior do projecto foi introduzido na equipa o Jorge Sousa, que fez o papel de consultor e também fez parte da equipa de desenvolvimento.

Inicialmente foram definidas certas regras como reuniões semanais da equipa com actas das mesmas, a construção de um site interno de projecto que albergasse uma introdução e documentação produzida e os alguns objectivos a alcançar em determinadas datas. Para além disto decidiu-se, embora sem data marcada, que ocorreriam duas apresentações relativas a este projecto: uma interna, nas instalações do Porto e uma nas instalações das Torres Vodafone em Lisboa, integradas na chamada *reunião de staff*<sup>3</sup>.

---

<sup>1</sup> General Design – documento que fornece uma visão global de cada aplicação produzida, a sua arquitectura e o seu fluxo

<sup>2</sup> Detail Design – documento que proporciona uma visão detalhada sobre a solução, incluindo código.

<sup>3</sup> Reunião de staff – reunião que ocorre periodicamente composta dos vários chefes das várias direcções e projectos existentes na Vodafone. As decisões importantes costumam ser tomadas nestas reuniões.

Foi decidido que uma vez o projecto piloto (fase de desenvolvimento) pronto, se procederia a uma migração para as máquinas da Nexenter<sup>4</sup>, que possuem uma instância do *iPlanet Portal Server* em produção e uma vez aí, os testes necessários à validação do projecto seriam feitos. Mais pormenores de planeamento poderão ser encontrados no Anexo B.

### **Aplicações Internas Envolvidas**

Neste projecto foram envolvidas as aplicações WebParceiros e SFAWeb, aplicações Web destinada aos agentes Vodafone.

Mais informações sobre estas aplicações encontram-se em 3.3. e 3.4.

### **Ramificações do Projecto “Portal de Serviços”**

No decorrer do projecto foi decidido pela direcção que a solução de Validação do Hardware Cliente (solução descrita na secção 4.1) seria implementada na aplicação WebParceiros (implementação descrita na secção 4.2).

Estas ramificações do projecto inicial representaram um considerável esforço de desenvolvimento e foram decisivas no sucesso do estágio curricular.

---

<sup>4</sup> Nexenter - Empresa do grupo Vodafone Telecel S.A. que fornece serviços como gestão dos *data-centers*, redes, e segurança.

## 2 Tecnologias e Produtos Utilizados

Este capítulo destina-se a ilustrar as tecnologias e produtos transversais a todo o projecto. Sempre que for pertinente, as tecnologias específicas a certas partes do trabalho serão apresentadas nos respectivos capítulos, para uma maior compreensão dos contextos em que foram utilizadas.

Como já foi referido, um dos requisitos transversais a toda a empresa, e por inerência aos projectos desenvolvidos no seu seio, é a segurança. Por este facto a investigação tecnológica levada a cabo teve sempre essa preocupação em mente.

### 2.1 Tecnologias

#### JAVA

Linguagem de programação orientada por objectos que se propôs, aquando da sua criação, a juntar a facilidade de utilização à potência do C++ e alcançar a portabilidade entre várias máquinas.

Para alcançar a portabilidade desejada, o JAVA é compilado para uma linguagem intermédia (acima do código-máquina normalmente gerado por compiladores de outras linguagens) chamada *bytecode*. O programa gerado pode ser executado numa máquina virtual JAVA, que traduz o *bytecode* para código máquina.

Criada pela Sun em 1995 ganhou um forte apoio tanto académico (pela sua curva de aprendizagem rápida) como a nível empresarial, pela sua flexibilidade.

Com o advento da plataforma J2EE, em 2000, foi dado mais um passo na sua entrada no mercado empresarial, obrigando a sua principal concorrente, a Microsoft, a propor uma plataforma e linguagens com algumas características semelhantes.

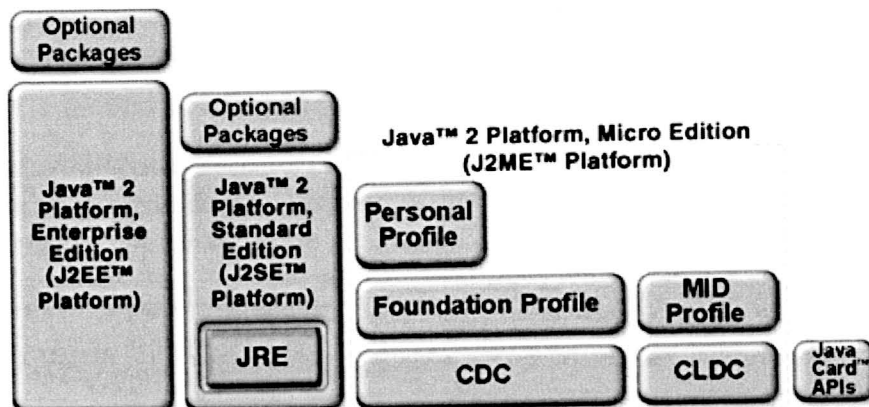


Fig. 2-1 Tecnologias - Estado actual da gama de plataformas JAVA

Entretanto houve extensões da linguagem, como sejam as servlets, JSP's, JDBC além do já referido J2EE. Essas tecnologias serão apresentadas em secções seguintes.

## J2EE

A tecnologia Java 2 Platform, Enterprise Edition (J2EE) fornece uma abordagem baseada em componentes ao desenho, desenvolvimento e entrega de aplicações empresariais. Esta plataforma oferece um modelo aplicacional distribuído e multi-ligado<sup>5</sup> (*multi-tiered*), possibilidade de reutilizar componentes, troca de dados baseada em XML, um modelo de segurança unificado e controlo de transacções flexível.

Uma característica importante desta tecnologia é que as soluções desenvolvidas não se encontram dependentes de nenhum produto específico; o consumidor final pode escolher o produto que melhor se adapte aos seus requisitos.

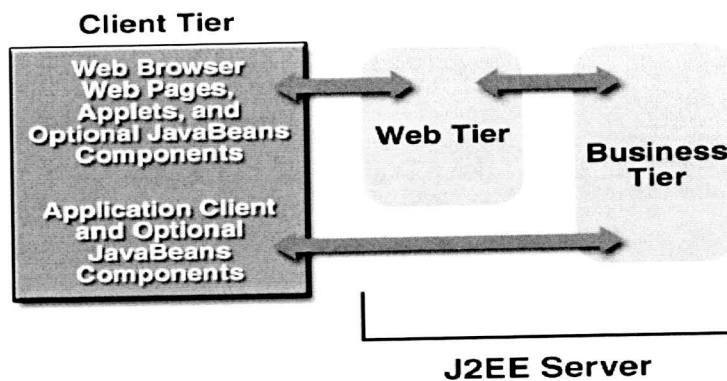


Fig. 2-2 Tecnologias - Arquitectura típica J2EE

### Componentes J2EE

As aplicações J2EE são constituídas por componentes. Um componente J2EE é uma unidade de software funcional e específica que é depois integrada numa aplicação J2EE e comunica com outros componentes. A especificação J2EE oficial define os seguintes componentes:

- Clientes aplicacionais e *applets* são componentes que correm no cliente.
- Java Servlet e JavaServer Pages (JSP) são componentes Web e correm no servidor.
- EJB's (Enterprise JavaBeans) são componentes de negócio que correm no servidor

Os componentes J2EE são escritos na linguagem de programação JAVA e são compilados como qualquer programa nessa linguagem. A diferença é que existe é no ciclo de execução a seguir à compilação, que consiste em:

- Verificação da validade formal e compatibilidade com a especificação oficial.
- Assemblagem numa aplicação J2EE.
- Entrega para produção onde são executados e geridos por um servidor J2EE.

<sup>5</sup> Multi-Tier – arquitectura genérica em camadas, cujo intuito principal é separar as aplicações em partes lógicas (camada de negócio, camada de apresentação, etc.) para que se possam desenvolver e alterar em separado sem grandes problemas. Existem extensões baseadas nesta arquitectura como a three-tier, model-view-controller, etc.



## EJB

Os EJB's são componentes Java reutilizáveis que implementam lógica de negócio e permitem desenvolver aplicações empresariais distribuídas e baseadas em componentes. Os EJB's residem num contentor (*EJB Container*) que fornece serviços padrão como persistência, segurança, transacções e concorrência, sendo assim totalmente portáteis para qualquer servidor que seja compatível com as normas definidas pela Sun.

### Componentes de um EJB's

- **Interface remota**, que expõe a lógica de negócio ao cliente
- **Interface local (home interface)**, que os clientes usam para criar, localizar e apagar instâncias EJB
- **Classe Bean**, que implementam a lógica de negócio

### Modo de Funcionamento de um EJB

Criação de um EJB:

- Codificação da lógica de negócio da aplicação numa classe.
- Especificação dos parâmetros de entrega (*deployment*) num *deployment descriptor*.
- Acondicionamento do EJB num ficheiro JAR<sup>6</sup>.
- Configuração do servidor aplicacional para carregar e executar esse EJB

A partir daqui, as aplicações clientes conseguem localizar o EJB e invocar os métodos definidos na interface remota.

## Servlets

Uma servlet é uma classe JAVA que corre num servidor compatível. As Servlets HTTP são um caso especial que recebem e manipulam um pedido HTTP e fornecem uma resposta HTTP, normalmente na forma de uma página HTML.

As Servlets HTTP têm acesso a todas as interfaces de programação do JAVA, incluindo todas as classes do JDK, residindo aí uma das suas maiores vantagens, já que transporta para o domínio das aplicações Web o poder de uma linguagem de programação completa e flexível.

## JSP (Java Server Pages)

As JSP são páginas dinâmicas para a Web, especificadas pela Sun Microsystems que combinam HTML com código Java.

Ao criar conteúdo dinâmico, é mais conveniente escrever JSP's do que servlets HTTP, porque no primeiro caso integra-se código Java directamente nas páginas HTML, ao contrário das servlets que devem fazer output de código HTML dentro do código em Java.

---

<sup>6</sup> Ficheiro JAR – ficheiro que empacota e opcionalmente comprime outros ficheiros, normalmente usado em classes JAVA

## LDAP

O LDAP (*Lightweight Directory Access Protocol*) é um protocolo para aceder e gerir um serviço de directório na Internet.

Um serviço de directório é uma aplicação distribuída de base de dados cujo fim é o de gerir entradas e atributos num directório, tornando-os disponíveis a outros utilizadores e aplicações.

Um directório é constituído por entradas que contém informação descritiva. Esta informação é armazenada em atributos de uma dada entrada. Um atributo pode ter mais do que um valor, e pode mesmo chegar a conter dados binários (p.ex: uma imagem).

Um serviço de directórios é parecido com uma base de dados, mas tende a guardar valores mais descritivos. Além disso, a informação é muito mais vezes lida do que escrita. Como tal, não é necessário implementar modelos complexos de transacções.

O LDAP define um protocolo de acesso a um serviço de directórios. Este protocolo é baseado na arquitectura cliente-servidor e corre sobre TCP/IP.

### Organização de directórios nos servidores LDAP

Sendo um serviço global de directórios, os dados no LDAP encontram-se organizados hierarquicamente, começando no topo e dividindo-se em ramos. Ao nível mais alto da hierarquia, as entradas representam grandes organizações e abaixo destas os ramos conduzem a organizações mais pequenas, podendo acabar finalmente em indivíduos ou recursos específicos.

Cada entrada é representada unicamente por um *distinguished name* (DN). Um DN é constituído por um nome que identifica essa entrada como sendo única naquele nível hierárquico, e por um caminho de nomes que traçam o caminho desde essa entrada até ao topo, ou *root*.

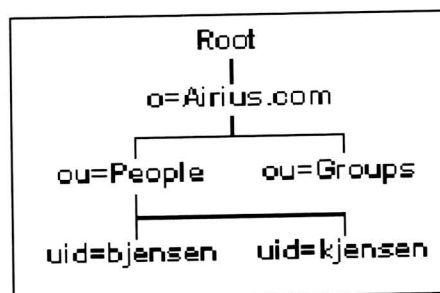


Fig. 2-3 Tecnologias - Exemplo de directório LDAP

Neste caso, “uid” representa o id único da entrada, “ou” representa a unidade organizacional a que a entrada pertence, e finalmente “o” representa a entrada de mais alto nível a que pertence.

Os dados são representados no formato de (*atributo:dados*). No caso anterior o *Distinguish Name* corresponde a “dn:uid=bjensen,ou=People,o=airius.com”.

A figura 2-4 ilustra a situação de uma entrada LDAP com mais atributos para além dos que constituem o DN:

```
dn: uid=bjensen, ou=people, o=airius.com
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: inetOrgPerson
cn: Babs Jensen
sn: Jensen
givenName: Babs
givenName: Barbara
mail: bjensen@airius.com
```

Fig. 2-4 Tecnologias - Entrada LDAP com múltiplos atributos

Pode-se dizer ao directório quais os atributos permitidos ou obrigatórios numa dada entrada, definindo classes de objectos para a entrada. Para se usarem classes de objectos é necessário primeiro identificá-las ao directório, por meio de um conjunto final de classes de objectos e atributos conhecidos como esquema (*schema*).

Para tal a verificação de esquemas no servidor deve estar ligada, permitindo que o servidor verifique que os atributos obrigatórios de uma dada classe existem, e que não existem atributos proibidos no directório. Em todo o caso é aconselhável não alterar muito o esquema base. Caso esse esquema seja demasiado limitador, pode ser melhorado com o uso de classes. No entanto, mudar completamente o esquema já existente pode levar a problemas de interoperabilidade entre diferentes servidores LDAP.

Os dados armazenados num directório LDAP podem ser distribuídos por outros directórios LDAP. Por exemplo, um servidor LDAP localizado em Airius.com pode conter entradas representando unidades organizacionais e indivíduos da América do Norte, enquanto que outro servidor LDAP pode conter as mesmas entradas, com dados referentes à Europa. Alguns servidores LDAP estão definidos para redireccionar pedidos para outros servidores LDAP. Por exemplo, se um servidor em Airius.com receber um pedido de informação acerca de um empregado no ramo Airius Corporation, este servidor pode redireccionar o pedido para o servidor LDAP no ramo Airius Corporation. Deste modo um conjunto de servidores LDAP pode aparentar ser um só directório, porque mesmo que não contenha a informação pretendida, o pedido pode ser redireccionado para o servidor que possua essa informação.

## CVS

O CVS<sup>7</sup> permite gerir um repositório de versões, tornando-se especialmente útil quando existem várias pessoas que podem modificar os mesmos ficheiros. Este sistema actua mantendo uma história de alterações, tendo cada uma dessas alterações atributos como o autor, a data, a hora e um comentário

Assim, este sistema permite responder a questões como:

- Quem fez determinada alteração?
- Quando foi feita?
- Porque foi feita?

## 2.2 Software Utilizado

### iPlanet Portal Server 3.0

#### **Arquitectura:**

O iPlanet Portal Server permite uma ligação segura entre uma máquina residente na Internet e uma intranet privada. Essa intranet está normalmente delimitada por um *router* e por *firewalls* que inibem o acesso público a áreas privadas e também o acesso privado a certas áreas públicas.

Uma intranet empresarial proporciona serviços importantes e necessariamente não-acessíveis ao comum dos utilizadores da Internet. No entanto, existem serviços que certos utilizadores (clientes, parceiros, empregados fora do local de trabalho) por vezes necessitam; neste contexto entra o uso de uma rede privada virtual, vulgo VPN<sup>8</sup>

Usar este software é uma maneira de proporcionar uma sessão virtual segura usando a rede pública (Internet). Neste aspecto o iPlanet Portal Server implementa uma VPN mas com algumas vantagens em relação às VPN's normais, a saber:

- Não é necessária configuração a nível de cliente, usado por exemplo em redes PPTP.
- Está disponível a todos os clientes via *browser web* compatível.
- Ao contrário das redes baseadas em *IPsec* em que a cifragem de dados se encontra ao nível 3 do modelo OSI, com este produto a cifragem encontra-se a nível aplicacional; assim, apenas necessita de haver confiança a nível de *browser Web*.

---

<sup>7</sup> CVS - *Concurrent Versions System*

<sup>8</sup> VPN – Rede Privada Virtual (*Virtual Private Network*) é uma rede privada que usa as infra-estruturas públicas de telecomunicações (p.ex. Internet) para proporcionar acesso seguro de indivíduos à sua rede empresarial. A segurança é conseguida usando protocolos que criam túneis encriptados de dados; deste modo, o emissor cifra e só o receptor pode decifrar os dados.

Tipicamente a configuração usada é a apresentada na figura 2-5.

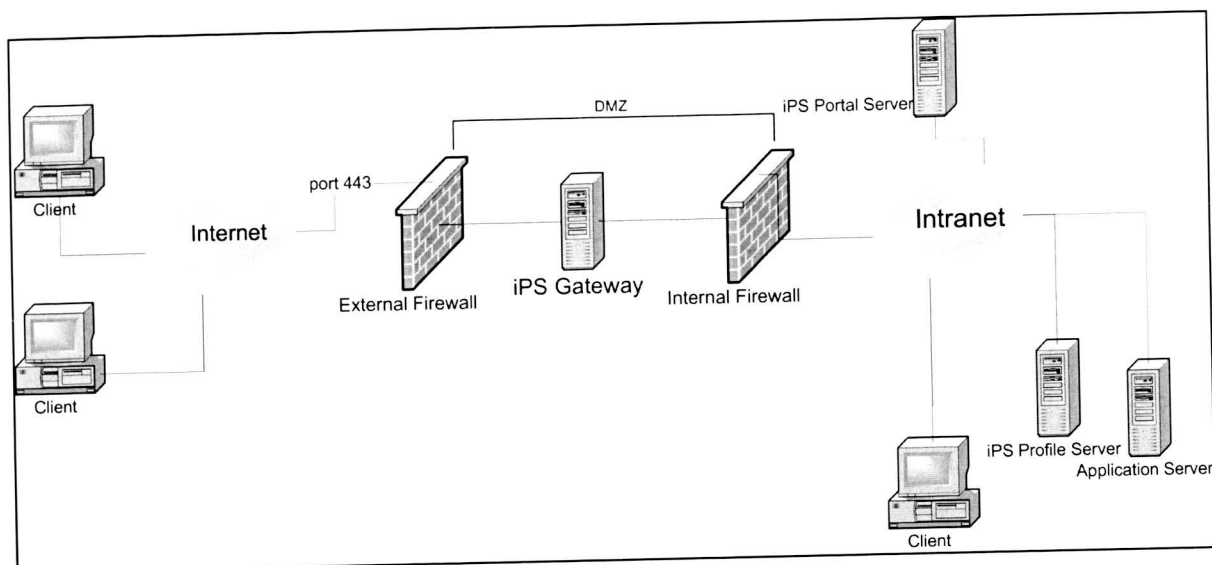


Fig. 2-5 Software Utilizado - Arquitectura do iPlanet Portal Server

Existe um componente, o iPortal Server Gateway, que reside numa zona desmilitarizada (*DMZ*<sup>9</sup>); esse componente serve de ponte segura entre o browser do cliente (HTTPS) e o servidor iPortal Server, que se encontra dentro da intranet da empresa. Por sua vez, este servidor comunica com os recursos que se querem tornar acessíveis, tais como servidores aplicativos e servidores Web.

As funções mais importantes desempenhadas por este componente são:

- Autenticação: a autenticação dos utilizadores nos diferentes modos disponíveis é feita por este componente. Entre os modos disponíveis encontram-se o LDAP, UNIX, Domínios Windows NT, *Safeword*, *RSA SecurID* e certificados digitais.
- Reescrita de endereços: modificação em tempo real do código HTML e JavaScript de modo a rectificar os endereços, juntando-lhes o endereço da própria gateway e forçando assim o browser a aceder aos conteúdos passando por este componente.
- Restrições: apenas os endereços, domínios e servidores configurados na administração são acessíveis. Assim, evita-se o caso de, depois de criada uma sessão válida na gateway, se aceder a um endereço qualquer na rede interna.

Para a autenticação e configuração do acesso dos utilizadores existe um servidor LDAP que é usado pelos dois componentes para obter informações. A componente iPlanet Profile Server é fornecida com o produto e é obrigatoriamente usada. Existem outro tipo de dados que podem residir noutro servidor LDAP, como utilizadores/*passwords* e dados de configuração.

<sup>9</sup> DMZ – Zona Desmilitarizada (*De-militarized Zone*) - Consiste em duas firewalls, chamadas externa (para a rede pública) e interna (para a rede privada). Assim o acesso ao componente de rede que lá reside está protegido de ambos os lados.

### Arquitectura baseada em papéis (roles)

Esta arquitectura consiste numa árvore de papéis em que o(s) domínio(s) gerais (ex: Vodafone) estão no topo. Descendo nessa árvore temos papéis atribuídos a grupos de utilizadores e, por fim, a utilizadores específicos.

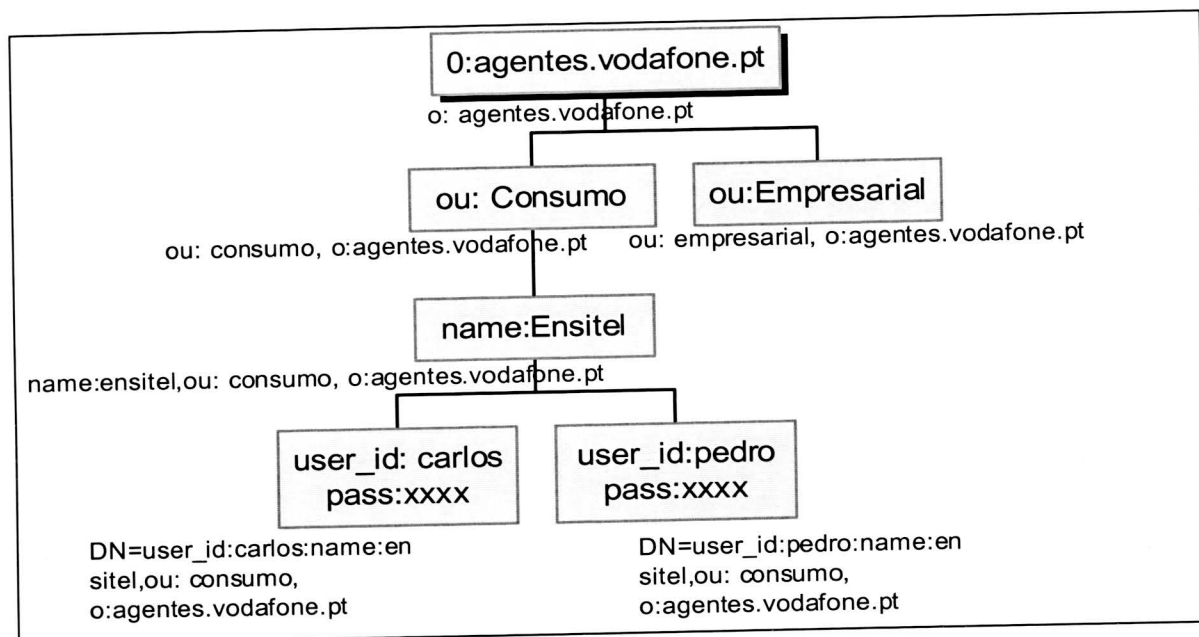


Fig. 2-6 Software Utilizado - Exemplo de árvore de papéis baseada em LDAP

Estes papéis possibilitam as seguintes funcionalidades:

- Configurar áreas de trabalho de utilizadores diferentes relativamente à disposição e a conteúdos.
- Definir políticas de controlo de acesso, a lista de aplicações que um utilizador pode usar e URL's a que pode aceder.
- Definir o tipo de autenticação usada (LDAP, NT), qual e o tempo máximo de sessão.
- Aceder a aplicações integradas no ambiente iPlanet Portal Server.

### Carga Balanceada

Numa configuração com várias gateways pode usar-se um balanceador de cargas conseguindo assim um ponto de entrada comum e deixando a cargo do balanceador o reencaminhamento de tráfego pelo melhor caminho.

### Single Sign-On

Uma vez autenticado, o utilizador nunca mais terá que fornecer qualquer userID e *password* durante a vida da sessão<sup>10</sup> dele no servidor, podendo aceder a aplicações preparadas para isso.

<sup>10</sup> Sessão – conceito aplicado nos servidores Web em que é atribuído um número único a um utilizador (baseado no *browser* utilizado) e permite ao servidor ter um espaço daquele utilizador para guardar informações e manter estados entre navegação



## Interfaces de programação

As interfaces seguintes são fornecidas para integrar uma qualquer aplicação na moldura aplicacional do iPlanet Portal Server:

- *Access Control and Profile API*: Serve para aceder a informações de perfil por parte de uma aplicação.
- *Session API*: Serve para as aplicações validarem a sessão e acederem a dados da mesma para um qualquer utilizador.
- *Logging API*: Proporciona um serviço de escrita, leitura e modificação de *logs*.
- *Content Provider API*: Fornece métodos para integrar conteúdo aplicacional externo no desktop do iPlanet Portal Server.
- *Pluggable Authentication API*: Permite escrever módulos adicionais de autenticação em Java.

No caso do uso do Java como linguagem de programação da aplicação, a comunicação com o iPlanet Portal Server é facilitada pois são fornecidas classes que podem ser usadas e que tornam a camada de comunicação transparente.

## Netlet

Fornece uma ligação segura entre um qualquer cliente num sistema com um browser que suporte JAVA, e um recurso de rede atrás de uma "firewall" empresarial.

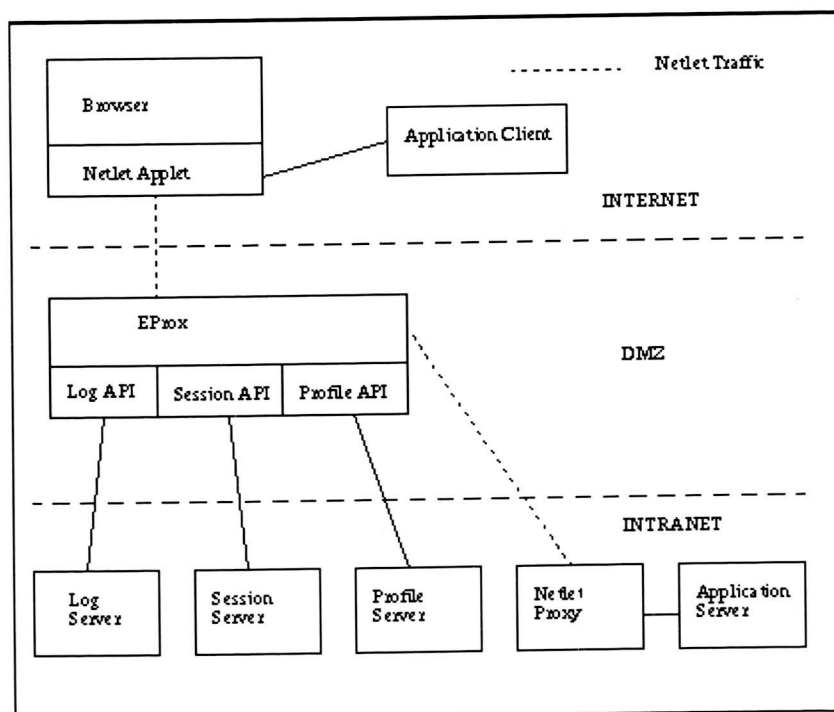


Fig. 2-7 Software Utilizado - Arquitectura da Netlet

O netlet consiste numa applet JAVA que cria um túnel de dados cifrado entre a máquina cliente e uma máquina denominada *Netlet Proxy*. A partir daqui é possível, por exemplo, configurar na máquina cliente um programa de correio electrónico para aceder ao correio corporativo, cujo servidor reside na intranet empresarial.

## BEA Weblogic 5.1

O servidor aplicacional BEA WebLogic é uma plataforma de comércio electrónico, que permite criar e disponibilizar rapidamente aplicações seguras, escaláveis e de fácil gestão. Este produto implementa a tecnologia J2EE, da Sun. Os detalhes “baixo nível” do sistema são geridos transparentemente pelo servidor, sendo deixada ao arquitecto a preocupação com a lógica de negócio e a apresentação. Este produto é o componente “Application Server” da figura 2-5.

### Arquitectura do Servidor Aplicacional WebLogic

A arquitectura típica do BEA Weblogic 5.1 baseia-se numa interpretação da arquitectura *multi-tier* e é composta por três camadas:

- a camada do cliente compreende os programas que acedem às aplicações (browsers Web, aplicações Web e aplicações *standalone*<sup>11</sup>)
- a camada do meio, que contém o próprio servidor aplicacional
- a camada de bastidores, que é composta por aplicações legadas, sistemas de gestão de base de dados e aplicações de suporte ao planeamento de recursos (ERP, *Enterprise Resource Planner*), etc.

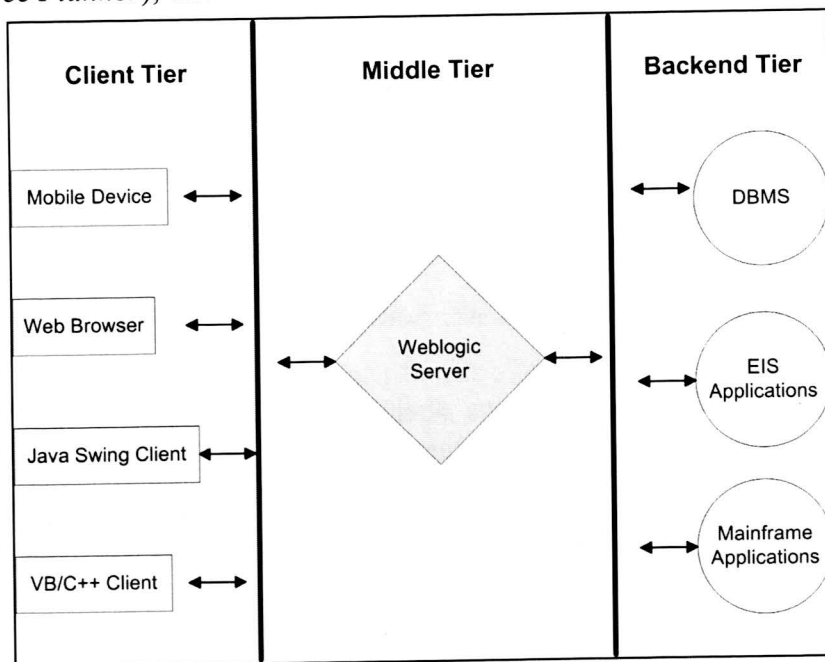


Fig. 2-8 Software Utilizado - Camadas do BEA Weblogic 5.1

### Vantagens

Algumas vantagens no uso dum servidor deste tipo são as seguintes:

- Implementação de *standards* abertos: a plataforma J2EE é um *standard* criado pela Sun acessível e implementável por qualquer pessoa.

<sup>11</sup> Standalone - Aplicações isoladas

- Portabilidade: as aplicações feitas podem facilmente ser migradas entre os vários servidores aplicativos existentes no mercado.
- Robustez: com a sua arquitectura bem definida e modular reduz-se a hipótese de erros
- Escalabilidade: podem ser acrescentadas componentes e aplicações, mas também modificar a apresentação gráfica de uma aplicação sem modificar a lógica de negócio já implementada

Como contrapartida, existe a necessidade de planear e estruturar muito bem toda a aplicação sob pena de se perder em performance ou em escalabilidade.

### Oracle 8i (8.1.7)

Trata-se de um SGBD (Sistema de Gestão de Base de Dados) relacional utilizado por milhares de empresas um pouco por todo o mundo. A descrição completa das características deste popular software sai do alcance deste relatório; a utilização feita deste SGBD neste projecto limitou-se no uso dos drivers JDBC<sup>12</sup>, consulta e modificação de tabelas, criação de conjuntos de tabelas e programação em PL/SQL<sup>13</sup>

### Sun ONE Directory Server 5.1

Este produto disponibiliza um repositório LDAP, e constitui um dos componentes fundamentais da arquitectura Sun ONE.

À medida que as empresas começam a utilizar a *Internet*, torna-se necessário gerir os potenciais utilizadores que irão usar esses mesmos serviços. Neste contexto, a solução deve ser flexível e escalável, a nível da gestão de utilizadores e mecanismos de segurança. Pode-se então armazenar e gerir perfis de utilizadores, bem como mecanismos de autenticação ao longo de todo um sítio *Web*.

As principais vantagens do *Sun ONE Directory Server 5.1*, em relação a um servidor LDAP comum são:

- Elevada disponibilidade e escalabilidade, suportando todo um negócio, à medida que cresce para milhões de utilizadores, e centenas de aplicações.
- Gestão de segurança e acessos, através de ferramentas próprias para gerir os utilizadores, os seus dados e as políticas que definem os privilégios de acesso. Potencia-se a privacidade do cliente na maior parte das aplicações de comércio electrónico, ao mesmo tempo que se torna possível introduzir segurança adicional (como certificados digitais) noutras aplicações mais sensíveis.
- Integração com outras fontes de dados dentro de uma empresa, permitindo disponibilizar informação já existente, com melhor qualidade de serviço e custos de administração mais reduzidos.

<sup>12</sup> JDBC - *Java DataBase Connectivity* – conjunto de normas independentes que definem a maneira de ligar uma classe em JAVA e uma base de dados. Se um SGBD implementar essas normas terá que escrever código específico para o seu produto e fornecer *drivers*, tal como acontece com o Oracle

<sup>13</sup> PL/SQL – linguagem de programação; permite desenvolver programas que executam directamente na Base de Dados utilizando código SQL e a maioria dos conceitos de programação, como sejam o controlo de fluxos, funções, procedimentos, etc.

- Dispõe de uma *interface* de programação (em C e Java) que permite ainda melhor integração com aplicações já existentes.

## Apache ANT

O Ant é uma ferramenta de desenvolvimento multi-plataforma e extensível desenvolvida em JAVA. O conceito base desta ferramenta é ser parecido com o *make*<sup>14</sup> mas sem as suas desvantagens. Os seus ficheiros de configuração são construídos em XML<sup>15</sup> e podem incluir comandos (chamados tarefas) como compilar, copiar, enviar através de ftp para o servidor de desenvolvimento, etc. Existem extensões ao ANT que adicionam tarefas que permitem, por exemplo, construir automaticamente um EJB (compilar, verificar, empacotar, copiar).

Em suma, é uma ferramenta que reduz o esforço posto na configuração e preparação do ambiente de desenvolvimento e permite a automatização de uma miríade de tarefas.

---

<sup>14</sup> make – ferramenta originalmente desenvolvida para o sistema operativo UNIX que controla a geração de executáveis a partir dos ficheiros de código fonte de um programa

<sup>15</sup> XML – eXtended Markup Language

## 3 Portal de Serviços

### 3.1 Sítio Web Interno do Projecto

As informações relativas a este sítio Web encontram-se no Anexo A.2 - Sítio Web Interno de Projecto.

### 3.2 Portal de Serviços

#### Introdução

Este Portal tem como objectivo proporcionar aos parceiros Vodafone (numa primeira fase), um acesso centralizado via *Internet* a aplicações sensíveis da empresa. Além disso (numa segunda fase) o Portal irá servir também como uma ferramenta para auxiliar o acesso de utilizadores de *Back-Office* às aplicações, dentro da própria Vodafone.

Todos os utilizadores dos parceiros, terão um *login* e uma palavra-chave própria. Adicionalmente, será configurado no servidor de perfis do iPlanet Portal Server quais as aplicações a que esses utilizadores poderão aceder. O acesso a essas mesmas aplicações será feito através de *Single Sign-On*. Isto significa que um utilizador só tem de inserir o seu *login* e palavra-chave uma única vez: quando entra no Portal. O acesso às restantes aplicações deve ser transparente.

Um requisito chave definido numa fase posterior do projecto foi que um dado utilizador (chamado NORMAL) não seja validado somente com o seu *login* e palavra-chave. É necessário ter a certeza que ele acede de um determinado PC, e que a esse PC foi dada permissão para aceder ao Portal. Assim, evita-se o acesso por parte, p. ex., de empregados do agente a partir de suas casas.

A um utilizador do tipo MASTER não é validada a máquina por onde acede, o mesmo acontecendo a um utilizador de *Back-Office*. Este último tipo acede directamente a aplicações *Back-Office*, quando aplicável.

Finalmente, e como meio de integrar este projecto numa possível migração para um ambiente mais centralizado, tornou-se necessário replicar para o directório de utilizadores do Portal os novos utilizadores criados nas aplicações de segundo nível.

#### Arquitectura Física do Projecto Piloto

A arquitectura do projecto piloto visa imitar ao máximo a implementação final da solução.

A diferença principal entre esta arquitectura e a arquitectura final é que neste caso todos os componentes se encontram fisicamente na mesma máquina. Adicionalmente, o ambiente final será constituído por máquinas com melhores desempenhos.

Neste projecto piloto usou-se uma máquina com as seguintes características:

- Sistema Operativo SunOS<sup>16</sup> 5.8

---

<sup>16</sup> SunOS - Sistema Operativo desenvolvido pela Sun Microsystems

- 512 Mb Memória
- Processador UltraSPARC-Iii

Esta máquina encontra-se fisicamente delimitada por duas *firewalls*, que marcam a zona desmilitarizada (DMZ), como se pode observar no diagrama da figura 3-1:

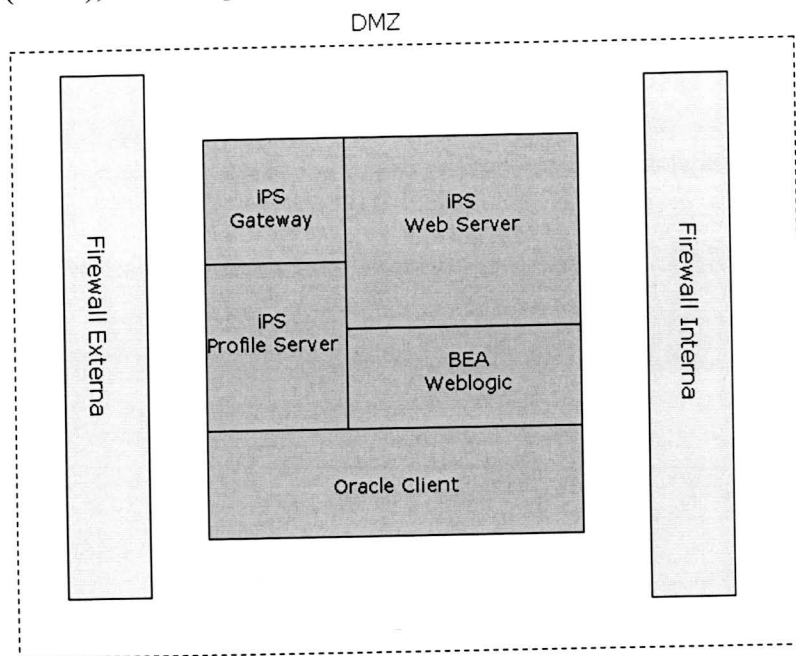


Fig. 3-1 Portal – Arquitectura do projecto piloto

A máquina Sun contém vários elementos essenciais para o funcionamento correcto do piloto:

- IPlanet Portal Server 3 Service Pack 4
- Bea WebLogic 5.1
- Oracle Client 8.1.7

Os componentes essenciais do Portal Server são:

- iPS Gateway
- iPS Web Server
- iPS Profile Server (directório LDAP)



## Configurações do iPlanet Portal Server

São várias as opções de configuração do iPlanet Portal Server. Nas próximas secções explicar-se-ão em mais pormenor as mais importantes.

### Administração do iPlanet Portal Server

O iPlanet Portal Server fornece uma consola de administração *Web* capaz de configurar a maior parte dos aspectos do Portal. Através desta consola são configurados domínios, utilizadores, papéis, permissões de acesso, mecanismos de autenticação e comportamento da componente *gateway*.

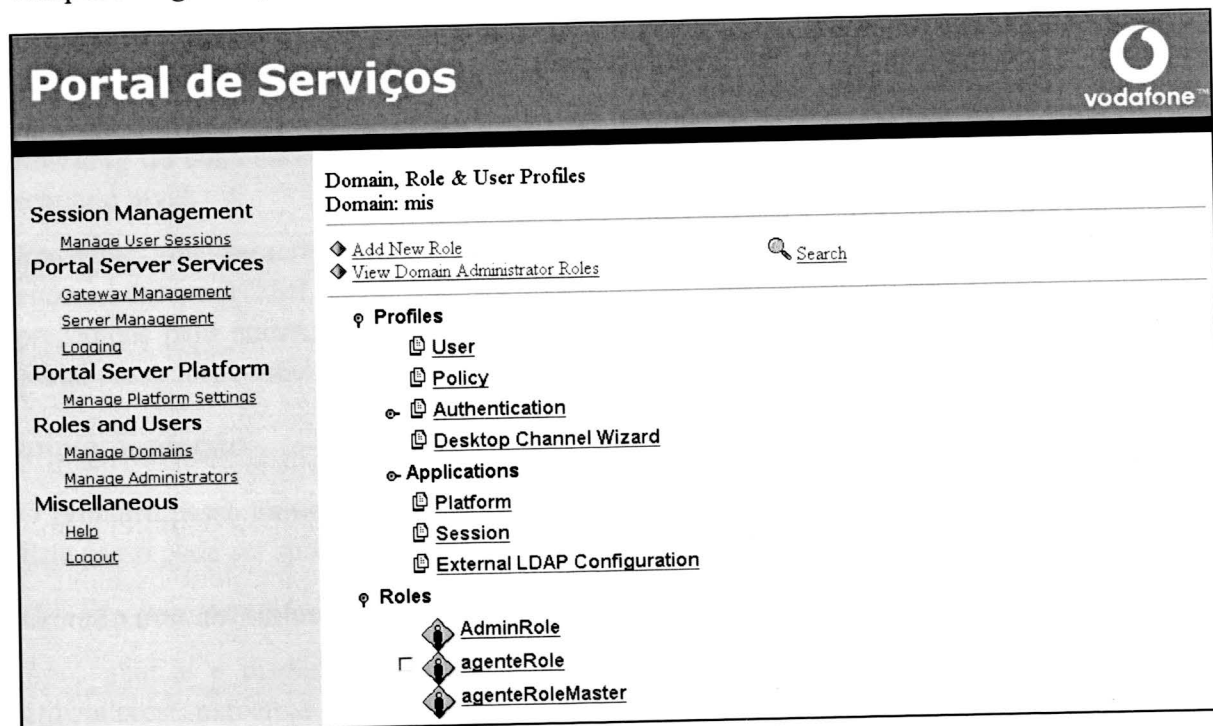


Fig. 3-2 Portal - Administração

### Domínios Disponíveis

O iPlanet Portal Server baseia o seu funcionamento em domínios. Cada domínio pode ter um conjunto de configurações própria, consoante o perfil dos seus utilizadores. A criação de um domínio é feita através da consola de Administração do iPS. Posteriormente, se for necessário mudar o aspecto das páginas *Web* iniciais do domínio, podem alterar-se os padrões HTML para esse mesmo domínio.

### Utilizadores e Papéis

Dentro de cada domínio disponível, existem um ou mais papéis. Um utilizador do Portal deve forçosamente pertencer a um papel, e consequentemente a um domínio.

Os vários atributos, políticas de acesso e configurações podem ser definidos para um domínio, papel (*Role*) e utilizador. O iPlanet Portal Server interpreta sempre os valores definidos a nível mais baixo, e, para os que não consegue encontrar aí, procura num nível superior.

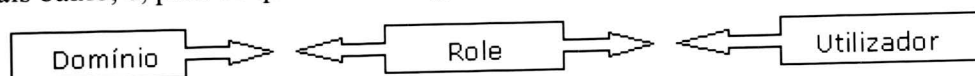


Fig. 3-3 Portal - Dependências entre Entidades do iPlanet Portal Server

## Gateway

Um dos componentes essenciais do iPlanet Portal Server é a *gateway*. Além de constituir o ponto de entrada comum no Portal, é responsável por enviar os pedidos HTTP dos clientes ao servidor *Web*, e recolher as respostas. Nesta fase aparece uma funcionalidade importante: *rewriting* do conteúdo dos ficheiros HTML que recebe do servidor *Web* e das aplicações de segundo nível.

*Rewriting*, ou reescrita, consiste em substituir todas as ocorrências e hiper-ligações encontradas no HTML resultante, de modo a que estas apontem para elas próprias, mas precedidas pelo endereço da *gateway*. Isto força que qualquer pedido feito pelo cliente ao portal passe forçosamente por este componente, reforçando a segurança.

## Implementação

O primeiro passo na implementação foi a configuração da interface do iPlanet Portal Server. Este passo pode ser subdividido em vários passos, descritos nas secções seguintes.

### Criação dos domínios

Após análise e discussão, decidiu-se criar domínios no Portal para estes dois tipos de utilizadores: parceiros e utilizadores *Back-Office*. Os parceiros pertenceriam ao domínio “mis”, e os utilizadores *Back-Office* ao domínio “ipsbackweb”. Para estes utilizadores o único pormenor que necessitam de saber para entrar no seu domínio, é o nome do mesmo. Por exemplo,

- <https://portal.servicos.pt/mis>
- <https://portal.servicos.pt/ipsbackweb>

### Criação de papéis e utilizadores

Foram criados alguns papéis e utilizadores dentro dos domínios definidos na secção anterior. No domínio “ipsbackweb”, criou-se um papel denominado “ipsBackRole”, que compreende todos os utilizadores de *Back-Office*.

No domínio “mis”, foram criados 3 papéis:

- “agenteRole” – compreenderá todos os utilizadores dos agentes Vodafone.
- “agenteRoleMaster” – albergará um utilizador que corresponde ao responsável por determinado agente. Este papel tem privilégios especiais, referidos mais adiante na secção.

- “adminRole” – compreenderá os utilizadores com privilégios de administrador de domínio (pode configurar papéis, políticas de acesso e aplicações, mas apenas no domínio a que pertence este papel).

### Configuração da Gateway

A configuração da Gateway foi feita de modo ao *rewriting* (falado acima) suportar várias funções JavaScript definidas nas aplicações de 2º nível.

Finalmente, para se poder utilizar as API's *Java* do iPlanet Portal Server numa máquina diferente daquela onde corre o servidor *Web* do Portal, é necessário enviar-lhe uma *cookie* contendo o estado actual da Sessão do utilizador no Portal. A lista de máquinas para as quais a *gateway* envia essa *cookie* tem de ser configurada previamente.

### Estruturação do directório LDAP

A estrutura criada teria de suportar:

- Utilizadores normais (MIS).
- Utilizadores com privilégios *back-office* (MISBack).
- Aplicações disponíveis para cada um deles.
- Informações acerca da máquina pela qual podem aceder.

Assim foi criada estrutura da figura 3-4:

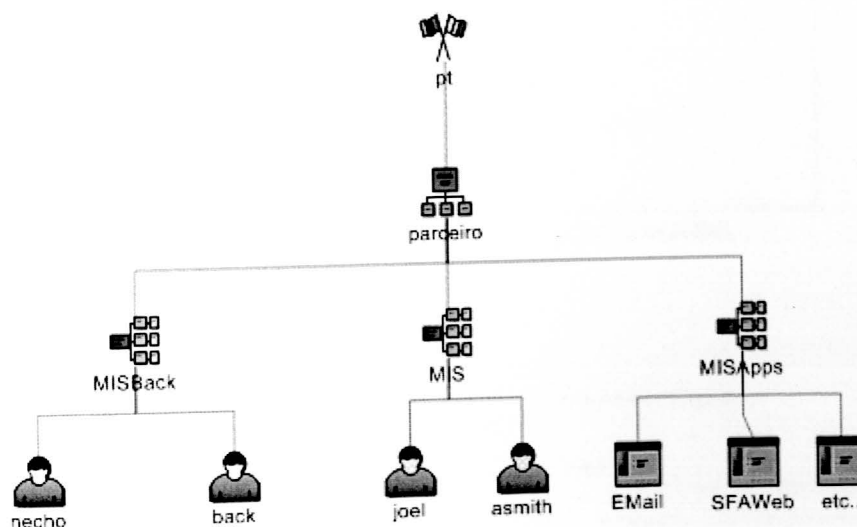


Fig. 3-4 Portal - Estrutura LDAP criada

### Novo módulo de autenticação

Este novo módulo teria que juntar dois tipos de autenticação:

- Autenticação LDAP, parte integrante do iPlanet Portal Server
- Autenticação da máquina que acede ao Portal

Para armazenar informações sobre as máquinas que podem aceder ao Portal foi desenvolvido um componente (chamado *iwtUserMacs*) que adiciona mais um opção na administração do iPlanet Portal Server. Essa opção pode ser vista na figura 3-5:

Fig. 3-5 Portal – Novo componente desenvolvido iwtUserMacs

Para a autenticação durante o *login* foi implementado o seguinte fluxo:

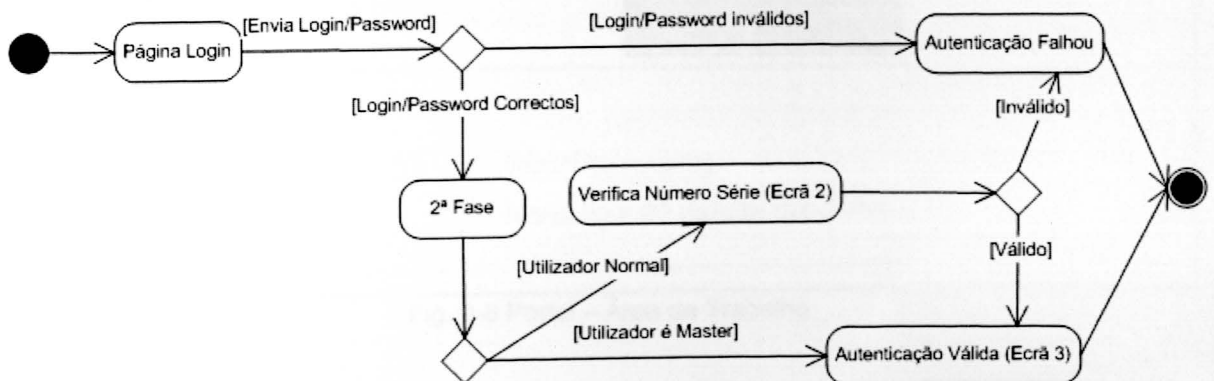


Fig. 3-6 Portal – Novo fluxo de *login*

Se o utilizador for normal é validado o número de série a seguir à validação do par (nome de utilizador, palavra-chave). No caso de um utilizador Back Office apenas é usado o módulo padrão LDAP.

**Interface Gráfica**

O ecrã inicial de autenticação do domínio mudou, como se pode verificar na figura 3-7:

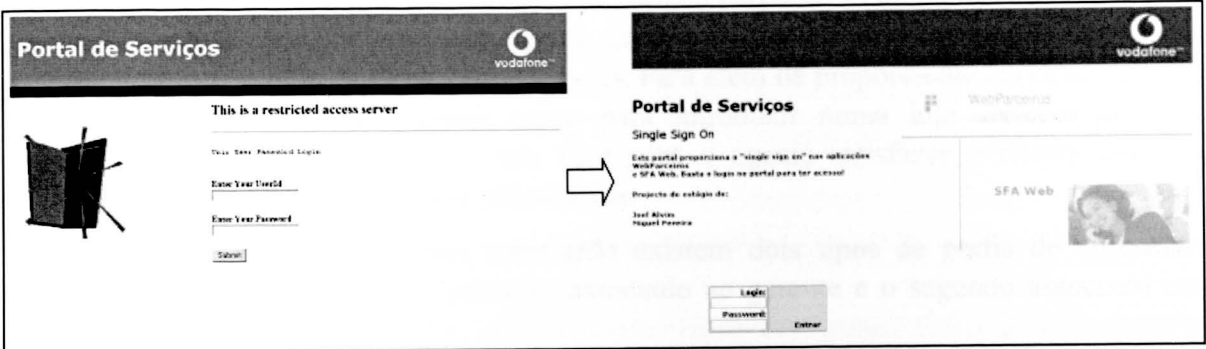


Fig. 3-7 Portal – Mudança na interface gráfica

A área de trabalho dos utilizadores NORMAL e MASTER é apresentada na figura 3-8:

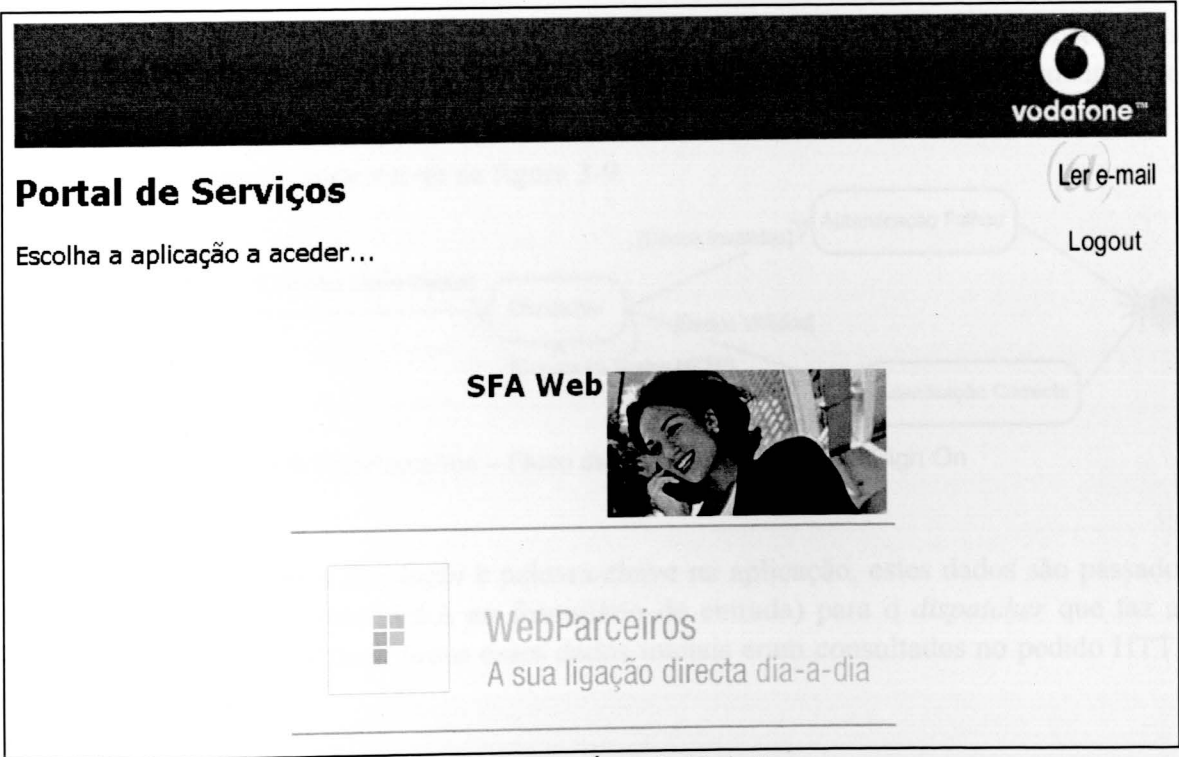


Fig. 3-8 Portal – Área de Trabalho

### 3.3 Aplicação Existente: Web Parceiros

#### Introdução

A aplicação WebParceiros existe para fornecer serviços úteis aos parceiros Vodafone, melhorando assim a eficiência e a qualidade da relação parceiro→Vodafone Telecel S.A. Um parceiro Vodafone define-se como um agente autorizado com um de vários graus de imagem (agente 100% Vodafone, agentes conjuntos, etc). Para além de proporcionar serviços de valor ao parceiro, esta aplicação também serve para introduzir numa loja serviços como o carregamento de Vitaminas, sendo mais fácil para o agente satisfazer o cliente que não possa/queira deslocar-se a uma caixa Multibanco.

Dentro de um determinado agente autorizado existem dois tipos de perfis de utilizador: MASTER e NORMAL, sendo o primeiro associado ao gerente e o segundo associado aos outros utilizadores.

#### Implementação do *Single Sign On*

A aplicação WebParceiros foi desenvolvida em JAVA (tecnologias JSP e EJB). Por via das interfaces de programação fornecidas, puderam alterar-se partes da aplicação para a tornar transparentemente integrada no Portal de Serviços e ao mesmo tempo utilizável como aplicação isolada.

A autenticação desta aplicação centra-se num componente denominado *dispatcher* (uma servlet JAVA), que centraliza as operações que compreendem o *login* (autenticação, preenchimento de variáveis de sessão, etc) e o *logout* (invalidação da sessão, etc) .

O fluxo de autenticação pode ver-se na figura 3-9.

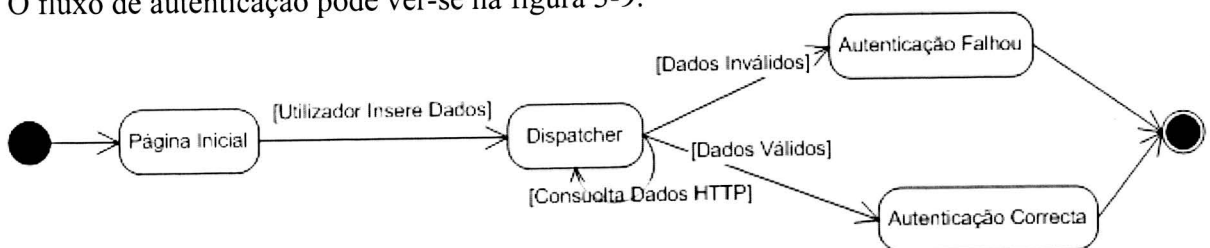


Fig. 3-9 WebParceiros – Fluxo de *login* antes de Single Sign On

Após o utilizador inserir o seu *login* e palavra-chave na aplicação, estes dados são passados (juntamente com outros escondidos no formulário de entrada) para o *dispatcher* que faz as operações de *login*. Nesta fase, todos esses dados iniciais eram consultados no pedido HTTP que chegava à *servlet*.

As modificações efectuadas a esse *dispatcher* são ilustradas no diagrama de fluxo da figura 3-10 e incluem uma autenticação dos dados recebidos usando o directório LDAP. Se essa autenticação for bem sucedida, o processo de *login* continua normalmente.

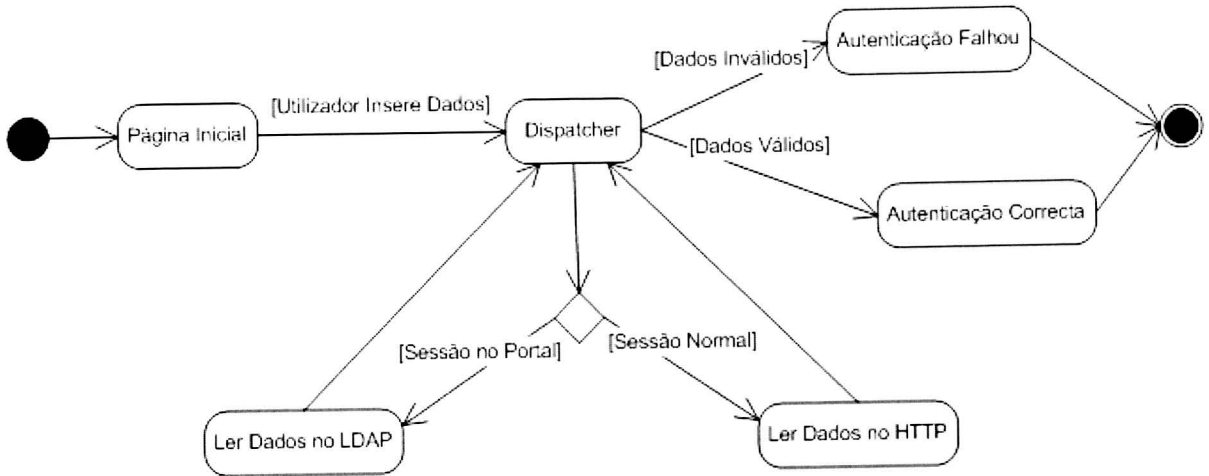


Fig. 3-10 WebParceiros – Fluxo de *login* com Single Sign On implementado

### 3.4 Aplicação Existente: SFAWeb

#### Introdução

A aplicação SFAWeb é usada pelos agentes Vodafone para acederem a informações de negócio incluindo as ofertas comerciais vigentes, os preços e descontos que estão autorizados a fazer, etc. Esta aplicação disponibiliza informações do sistema Siebel<sup>17</sup>, usando para esse efeito a interface *SiebelWeb*, configurável e baseada num servidor Windows NT. O acesso a código fonte desta aplicação é vedado.

#### Implementação do *Single Sign On*

A integração desta aplicação foi conseguida de um modo simples: por programação foi criada uma página que simulasse um *login* por POST. Esta página é criada por um adapter<sup>18</sup> *SFAWebAdapter* que retira a informação do utilizador presente no directório LDAP, cria uma página contendo campos de texto escondidos (HTML) e simula um POST, imitando o que um utilizador faz quando submete a informação de um formulário (HTML).

A informação é retirada de classes desenvolvidas especialmente para o efeito, como sejam a *Application* que representa uma aplicação de segundo nível como todos os seus atributos e a *LDAPMisWorker*, que serve de *interface* ao directório LDAP. O uso destas classes torna as integrações mais transparentes, como se pode ver no segmento de código seguinte.

```
String urlDest = app.getServerName();
```

<sup>17</sup> Siebel – Aplicação de CRM (Costumer Relationship Management) que interliga diversos sectores de uma empresa e facilita o fluxo de informação entre os departamentos que tratam com o cliente final.

<sup>18</sup> Adapter – classe genérica desenvolvida para servir como adaptador a uma aplicação. Para cada uma das aplicações existem depois classes-filhas que terão as informações específicas.



```
LdapMisWorker ldap = new LdapMisWorker(null, false);
String userPortal = pa.getSession().getClientID();
Hashtable userDetails = ldap.getUserAppDetails(userPortal, "sfaweb");
String username = (String)userDetails.get("login");
String password = (String)userDetails.get("pass");
```

Com este segmento retiram-se todas as informações necessárias à aplicação SFAWeb. A seguir gera-se a página propriamente dita e procede-se à submissão automática recorrendo à geração de código *JavaScript*. A aplicação SFAWeb está à espera de uma *cookie*<sup>19</sup> com o atributo “uName” (nome do utilizador) aquando do *login*, sendo esta *cookie* também construída. Depois de construída a *String* com estas informações, as mesmas são enviadas para o *browser*, ocorrendo assim a entrada na aplicação

```
String responseContent="
(...)
"
res.setContentType("text/html");
out.println(responseContent);
```

Para as palavras-chave dos utilizadores não serem guardadas em claro no directório LDAP utiliza-se um sistema de cifragem simétrica DES. A implementação usada em JAVA é fornecida por classes da SUN (JCE, *Java Cryptography Extension*) e só é usada para decifrar as palavras-chave. A cifragem é feita em PL/SQL aquando da replicação descrita em 3.6.

### 3.5 Aplicação Nova: Feedback Reports

A aplicação FeedbackReports foi uma ideia do Eng. Henrique Necho que teria três objectivos principais:

- Adicionar mais uma aplicação ao portal, de modo a demonstrar a sua escalabilidade.
- Proporcionar aos operadores que recebem chamadas relativas às aplicações WebParceiros e SFAWeb um modo de armazenarem certas características desse pedido.
- Fornecer estatísticas relativas a essas chamadas.

A descrição completa desta aplicação encontra-se no Anexo A.1 – Aplicação Feedback Reports

---

<sup>19</sup> Cookie – segmento de informação armazenado no lado do cliente e que guarda dados definidos no protocolo HTTP (como domínio, tempo de vida) e dados definidos programaticamente, como um nome de utilizador, id da sessão no servidor, etc. Este método é extremamente usado porque permite manter um estado durante a navegação num site Web.

## 3.6 Replicação de Utilizadores

### Introdução

Esta secção diz respeito a uma parte importante de qualquer solução *single sign-on*: a replicação de utilizadores.

Neste projecto desde cedo se percebeu que seria necessário uma solução que automatizasse a criação de utilizadores, já que existem várias aplicações no portal (referidas como “Aplicações de 2º nível”) a que queremos que se aceda transparentemente. Esta replicação torna-se ainda mais necessária porque as aplicações já existiam antes da introdução do Portal de Serviços, com todos os utilizadores criados e esquemas de autenticação bem definidos.

Para proporcionar então uma solução de *single sign-on*, as informações necessárias para o acesso dos utilizadores às várias aplicações de 2º nível terão de estar centralizadas, neste caso encontrando-se num directório LDAP. Numa primeira fase esta replicação diz respeito às aplicações já mencionadas: WebParceiros e SFAWeb. O projecto geral da solução que permite preencher esse directório LDAP será apresentado a seguir.

### Requisitos Iniciais

Os principais requisitos encontram-se presentes nesta frase, compilada a partir de actas de reuniões em que se discutiu a solução pretendida:

“Implementar uma solução tão segura quanto possível que seja viável e genérica o suficiente para alargar o espectro de aplicações de segundo nível num futuro próximo”.

Ficou decidido há algum tempo que a aplicação “dominante” seria o WebParceiros, ou seja, uma criação ou modificação de um utilizador no WebParceiros será forçosamente precedida da criação deste mesmo utilizador no Portal de Serviços (ou da modificação, se esse utilizador não existir). Na replicação referente à aplicação SFAWeb apenas existem modificações de utilizador, tendo esses utilizadores de já existir no âmbito do Portal de Serviços.

O mecanismo utilizado deve ser compatível com a replicação “forçada” dos utilizadores já criados, necessária para a população do directório LDAP numa primeira fase.

Deveriam ainda existir mecanismos para prevenir falhas, p.ex. uma fila de espera para transacções com recolha de erros em caso de falha. Esta fila de espera permitiria armazenar operações de replicação e posteriormente um outro programa consultá-la e executaria as operações necessárias.

### Arquitectura

Genericamente podem considerar-se neste momento 3 componentes principais nesta solução:

1. Processo que verifica a fila de espera.
2. Código de replicação adicionado à aplicação WebParceiros.
3. Código de replicação adicionado à aplicação SFAWeb.

O código adicionado deve interceptar qualquer operação de criação ou mudança de password de um utilizador e armazená-la na fila de espera, para que posteriormente essas operações sejam também executadas no directório LDAP.

Estes 3 componentes serão explicados com mais detalhe no capítulo “Descrição de Implementação”

### Arquitectura

A arquitectura lógica da solução encontra-se ilustrada na figura 3-11:

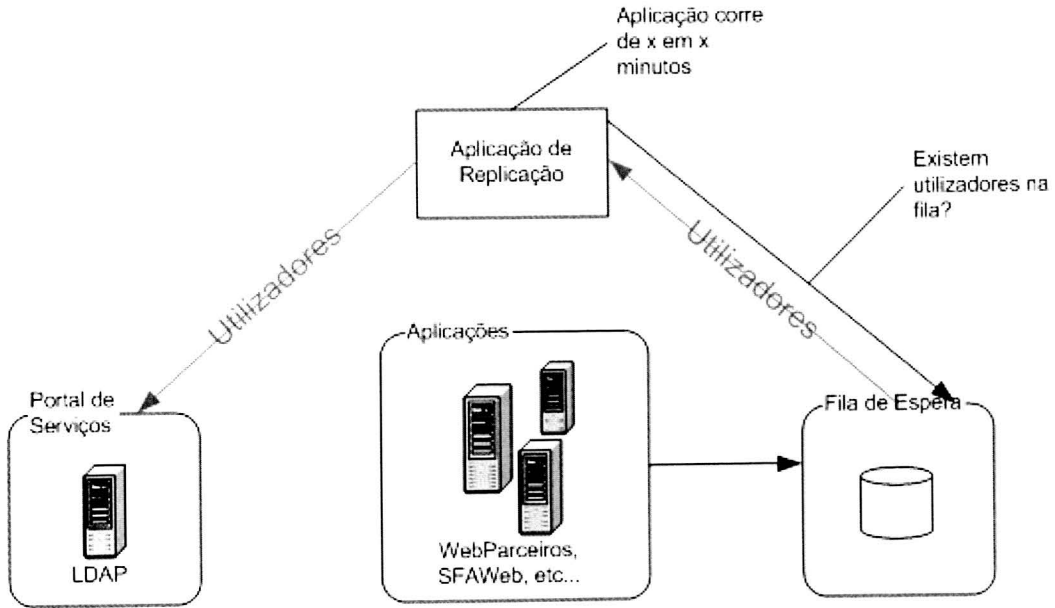


Fig. 3-11 Replicação - Arquitectura Lógica

### Arquitectura Física

Na arquitectura física existem duas opções principais de implementação:

- Criar na base de dados de cada aplicação de segundo nível a tabela que serve de fila de espera
- Haver uma base de dados que centraliza essa fila de espera.

No código presentemente implementado qualquer uma destas opções pode ser usada.

### Descrição da Implementação

#### Integração Genérica

Esta integração está a cargo de uma classe JAVA, `DbLdapReplicator`, que é executada como um processo em *background*, verificando a fila de espera e fazendo as mudanças indicadas no diagrama 3.12.

Fila de Espera

A tabela que implementa a fila de espera tem os seguintes campos:

LDAP REPLICATION

| Row # | ID | USER_APP     | PASS_APP                         | USER_PORTAL | PASS_PORTAL | STATUS | CREATION_DATE       | OPERATION | APPLICATION | ERROR |
|-------|----|--------------|----------------------------------|-------------|-------------|--------|---------------------|-----------|-------------|-------|
| 1     | 22 | AGT_SFawEB21 | 603EC342AE36E6E4A671620AA8BCE848 |             |             | PS     | 14-08-2002 22:47:51 | UPDATE    | sfaweb      |       |

Os campos têm o seguinte significado:

- ID : Gerado automaticamente
- USER\_APP : *Login* para a aplicação
- PASS\_APP : Password para a aplicação
- USER\_PORTAL : *Login* para utilizador a criar no portal (se for o caso do WebParceiros)
- PASS\_PORTAL : Password para utilizador a criar no portal (cifrado em SHA-1, padrão de cifragem do servidor LDAP utilizado)
- STATUS : Estado da transacção (PS se estiver à espera de ser executada, FA em caso de falha)
- CREATION\_DATE : Data da criação do registo
- OPERATION : INSERT ou UPDATE, não está a ser usado neste momento porque esta lógica encontra-se implementada na classe DbLdapReplicator. Este campo é apenas informativo.
- APPLICATION : A aplicação para a qual se vão fazer as alterações.
- ERROR : O eventual erro ao fazer as alterações.

Classe de Replicação

Na figura 3-12 encontra-se o diagrama de estados que ilustra o funcionamento da classe de replicação DbLdapReplication

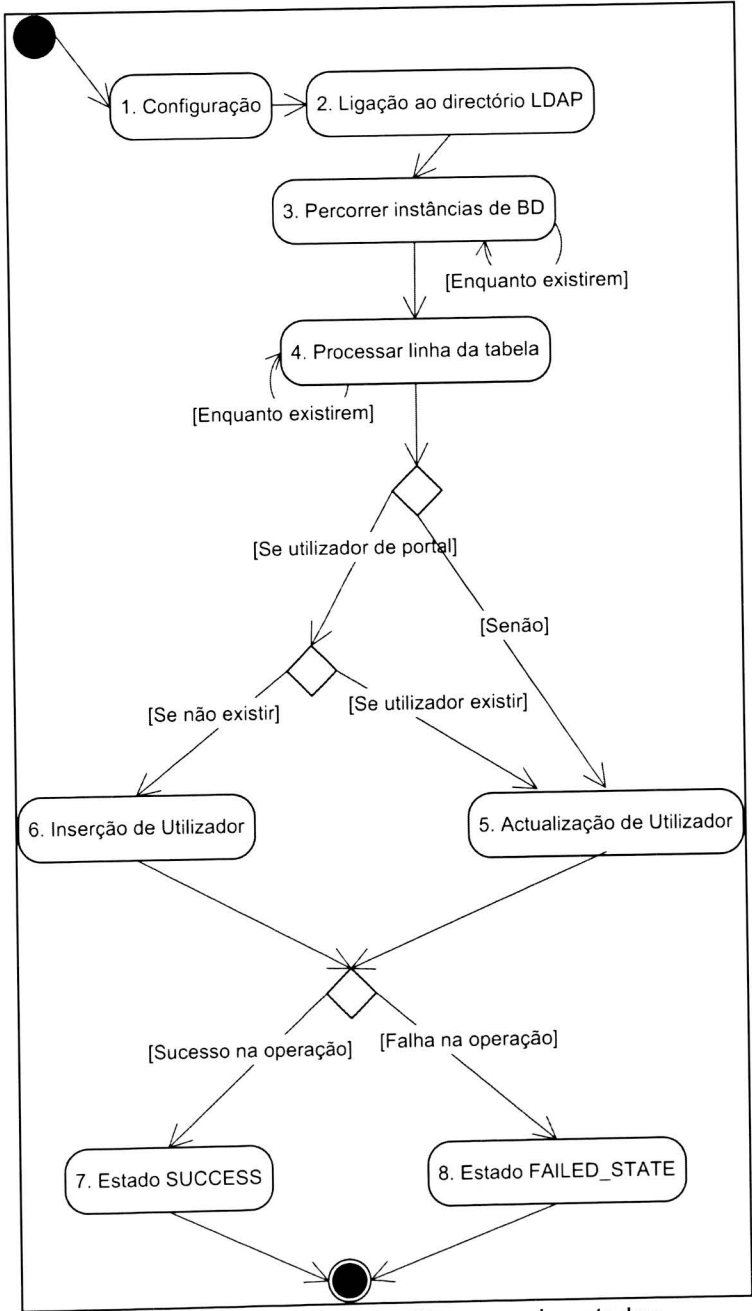


Fig. 3-12 Replicação - Diagrama de estados

O significado completo de cada um dos estados é explicado a seguir:

1. **Configuração** – Consiste na leitura de um ficheiro de propriedades onde se especificam, entre outras coisas:
  - 1.1. Lista de bases de dados que contém tabelas **LDAP\_REPLICATION**.
  - 1.2. Se a base de dados se refere à aplicação WebParceiros os utilizadores especificados são denominados “utilizadores de portal”, ou seja, serão criados no directório LDAP do iPlanet Portal Server se não existirem.
2. **Ligação ao directório LDAP** – auto-explicativo
3. **Percorrer instâncias de BD** – ciclo que percorre todas as instâncias lidas em 1.
4. **Processar linha da tabela** – esta operação lê e interpreta todas as linhas da tabela; passando para várias decisões
  - 4.1. Se o utilizador for de portal
    - 4.1.1. se não existir no directório LDAP passa para 6.
    - 4.1.2. se existir passa para 5.
  - 4.2. Se o utilizador não for de portal
    - 4.2.1. passa para 5.
5. **Actualização de utilizador** – Actualiza um utilizador no directório LDAP
6. **Inserção de utilizador** – Insere um utilizador no directório LDAP
7. **Estado SUCCESS** – Replicação bem sucedida, a linha da tabela é apagada
8. **Estado FAILED\_STATE** – Replicação falhou, a linha da tabela guarda o estado FAILED\_STATE e é guardado também o erro correspondente

### **Replicação WebParceiros**

Na aplicação WebParceiros foi necessário acrescentar um novo EJB e alguns segmentos de código nas páginas JSP que o utilizam em dois pontos chave:

- Activação de um utilizador
- Alteração de password

Estas alterações foram necessárias porque um utilizador do Portal de Serviços tem que ter a sua password cifrada em SHA-1, portanto nada se podia fazer com o *digest* MD5 da password guardada pela aplicação WebParceiros aquando das operações de registo e mudança de password..

EJB de replicação: PMSPUserReplicationEJB

Este *Enterprise Java Bean* serve para fazer a ponte entre a aplicação e fila de espera. A interface remota expõe apenas o método

```
insertReplicationUser(String user, String pass, String app, String digestAlgorithm)
```

que recebe um user, uma password, uma aplicação e um algoritmo de *digest* que serve para cifrar a password da aplicação.

Com estes dados é construído um registo e inserido a fila de espera, LDAP\_REPLICATION.

Ponto Chave de Modificação: Activação de um utilizador no WebParceiros

Na opção “Activar”, acessível na página de *login* da aplicação, é pedido ao utilizador o código de activação e a password que quer utilizar. Nesta alteração é usado o EJB descrito acima para inserir os dados na fila de espera.

Ponto Chave de Modificação: Alteração de password

Na opção do menu principal da aplicação “Utilizador → Password” é fornecida ao utilizador a hipótese de alterar a sua password. Nesta alteração é usado o EJB para inserir os dados na fila de espera.

### **Replicação SFAWeb**

Neste caso existem dois componentes:

- Ficheiro **.asp** a ser usado numa alteração de password
- *Stored Procedure* em PL/SQL

#### Script ASP

Este script chama a *stored procedure* em PL/SQL com a nova password, de modo a esta alteração entrar na fila de espera.

#### Stored Procedure PL/SQL

Esta stored procedure insere um registo na tabela **LDAP\_REPLICATION**. A palavra-chave é inserida num formato cifrado usando o algoritmo de chave simétrica DES, sendo posteriormente decifrado pela classe JAVA `DbLdapReplication` falado na secção acima.



## 4 Ramificações do Projecto Portal de Serviços

### 4.1 Validação do *hardware* cliente

Esta solução permite identificar de uma maneira segura uma máquina cliente que acede a um sítio Web. Foram investigados vários métodos, entre os quais o uso de *cookies*, o uso de PDC's<sup>20</sup> e o uso de uma característica única do *hardware* do cliente. Destes três métodos, o escolhido foi o terceiro, por ser o que melhor satisfaz os requisitos iniciais.

A solução foi originalmente desenvolvida apenas em JAVA e integrada apenas no Portal de Serviços. No seguimento da apresentação aos directores numa reunião de *staff* que incluiu uma apresentação por parte da equipa de desenvolvimento, foi decidido que a solução iria ser implementada na aplicação WebParceiros, essencialmente por medo de fraudes já que os utilizadores podem fazer carregamentos de telefones móveis a partir dessa aplicação. Com esta decisão tomada foi necessário refinar e adaptar a solução inicial, que, por precisar de um download de mais de 5MB (JVM, Java Virtual Machine), foi considerado demasiado custoso para o utilizador.

#### Requisitos Iniciais

Os requisitos subdividem-se em funcionais, requisitos que têm a ver com funcionalidades objectivas da solução e os não funcionais, que versam as diversas restrições impostas em relação, por exemplo, à facilidade de utilização, tempo dispendido, etc.

##### Funcionais

- Identificar univocamente uma máquina cliente que acede a um site
- Usar preferencialmente o endereço MAC<sup>21</sup> do adaptador de rede da máquina (melhor solução investigada)

##### Não Funcionais

- Segurança acima de tudo
- Instalação e execução o mais leve e automatizada possível
- Mínimo de interacção com a máquina cliente
- Evitar a todo custo interacção física com a máquina cliente (distribuição de CD's, visitas de técnicos para instalação, etc)

#### Tecnologias Relevantes

##### Applet JAVA

Esta tecnologia é parte integrante do JAVA e trata-se de uma classe escrita nesta linguagem que é descarregada e executada no cliente (tipicamente num *browser* Web), numa máquina virtual JAVA previamente instalada.

---

<sup>20</sup>Personal Digital Certificates – Certificados digitais que autenticam um cliente. Cada um destes certificados tem que ser comprado a uma entidade certificadora, como a Verisign, tornando-se uma solução segura mas cara.

<sup>21</sup>MAC – Medium Access Control – endereço gravado numa placa de rede que permite identificá-la univocamente.

## Relevância para o projecto

Juntamente com o JNI (descrito abaixo), uma applet JAVA devidamente certificada (assinada com um certificado de confiança emitido por uma entidade certificadora) pode aceder ao hardware e retirar as informações necessárias.

### JNI

O JNI (*JAVA Native Interface*) é uma interface de comunicação com o sistema operativo que permite ao JAVA usar métodos nativos e a programas noutras linguagens invocarem a JAVA Virtual Machine. Usando esta interface o JAVA perde portabilidade mas ganha mais versatilidade, podendo aceder ao hardware, aos serviços do sistema operativo, etc.

## Relevância para o projecto

Neste projecto é necessário validar a máquina cliente, acedendo para esse efeito ao hardware. O JAVA não fornece classes de acesso ao hardware, tendo de ser usada a API do Windows, através do JNI.

### ActiveX

O ActiveX é uma evolução das tecnologias COM (*Component Object Model*) e OLE (*Object Linking and Embedding*), da Microsoft. Uma das implementações desta tecnologia é o controlo ActiveX, um objecto que pode ser descarregado e executado automaticamente por meio de um *browser* Web compatível..

## Relevância para o projecto

Este projecto necessita de um programa que corra num cliente e aceda ao hardware de um modo automático, tal como o controlo ActiveX. Este controlo é autenticado usando um certificado de confiança emitido por uma entidade certificadora, que certifica o criador e distribuidor do controlo, necessitando o utilizador de autorizar a sua execução.

Para além disto, as operações de descarregamento e instalação são automáticas, necessitando o programador apenas de configurar o contentor<sup>22</sup> deste ficheiro.

### SNMP

O SNMP (*Simple Network Management Protocol*) consiste num protocolo simples que permite obter informações e gerir uma rede a partir de um só ponto. Para obter essas informações é necessário que haja entidades de nome “agentes” e “gestores”. Podemos imaginar o gestor como uma consola de administração e os agentes instalados em hardware como computadores, impressoras, servidores, routers, etc.

---

<sup>22</sup> Tipicamente um ficheiro CAB (Cabinet), comprimido e contendo todos os componentes necessários à execução do controlo ActiveX, como sejam bibliotecas, etc.

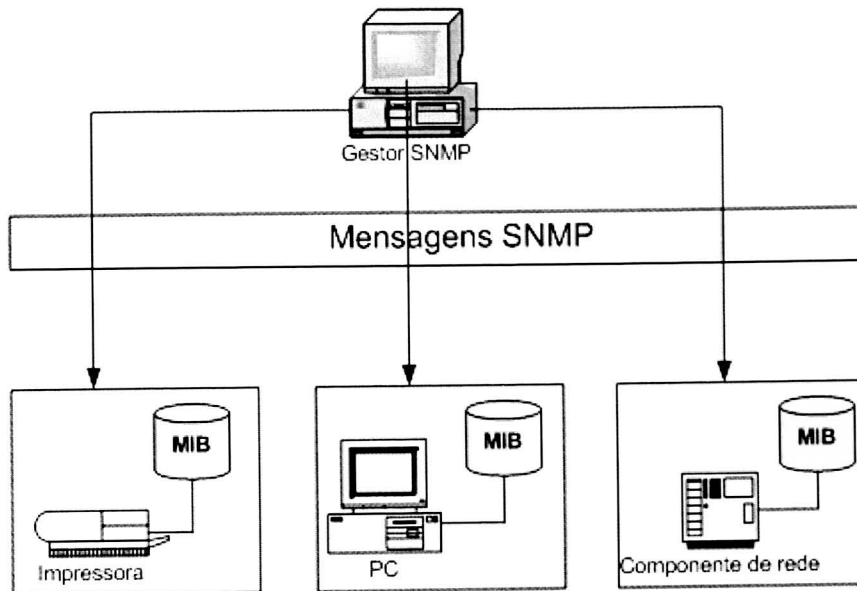


Fig. 4-1 Validação de Hardware - Arquitectura SNMP

Um agente contém uma MIB, *Management Information Base*, que define as operações que podem ser feitas, tanto de informação (GET) como de parametrização (SET). Esta base de informação contém objectos que têm um identificador único.

Este identificador contém, entre outras coisas:

- Tipo do objecto
  - Counter
  - String
  - Gauge
  - Address
- Nível de acesso
  - Leitura
  - Leitura/Escrita
- Restrições de tamanho
- Restrições de gama

Um agente pode comunicar assincronamente com o gestor, por meio de uma “trap” (mensagem não solicitada que informa de um certo estado).

### Relevância para o projecto

Este projecto usa o protocolo SNMP para obter o endereço MAC dos adaptadores de rede instalados.

Este método foi o escolhido porque é o que oferece maior fiabilidade e é o método padrão usado pelo Windows para obter informações (no utilitário de linha de comandos da Microsoft *ipconfig*, p.ex.). Além disso permite usar o mesmo código para todas as versões do Windows, necessitando apenas que tenha o subsistema de rede instalado (o que acontece quando se instala, p.ex., um adaptador de rede).

Um outro métodos estudado, o acesso pela API NetBIOS do Windows, implicava não só que se tivesse instalado as camadas de rede mas também que se tivesse activo o “File and Pinter Sharing” no computador, o que é usual mas pode não acontecer.

## Arquitectura

### Diagrama de Componentes

O diagrama da figura 4-2 ilustra os vários componentes lógicos que compõem o sistema.

As linhas a traço interrompido ilustram relações de dependência e as linhas a traço contínuo relações de comunicação.

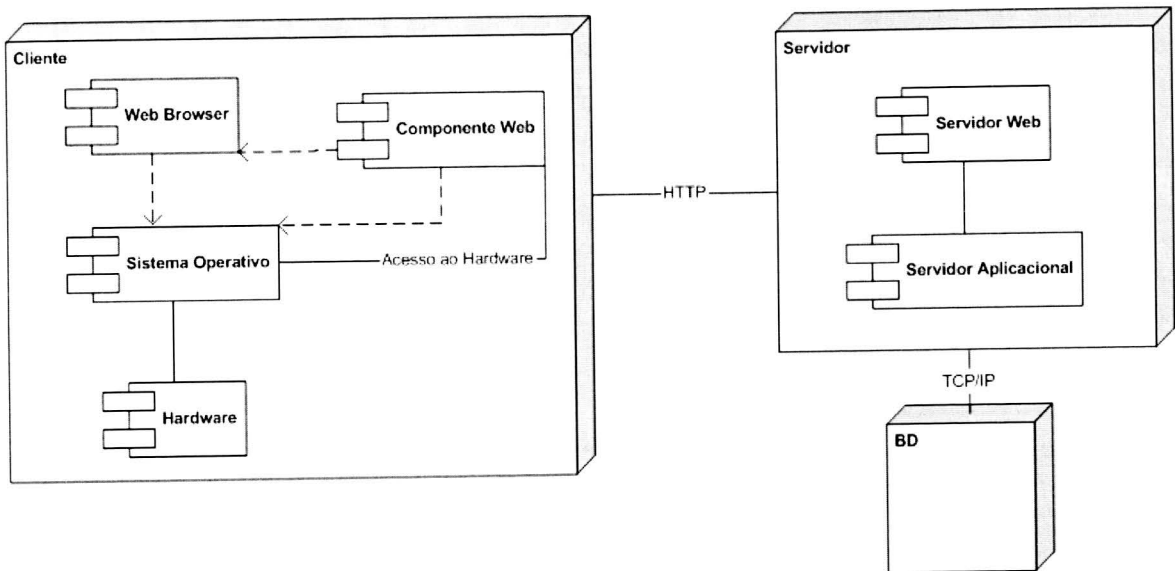


Fig. 4-2 Validação de Hardware - Diagrama de Componentes

### Revisão aos Requisitos Iniciais

A implementação seguiu os requisitos definidos no início do projecto, havendo no entanto algumas mudanças, que se deveram ao seguinte:

- Adopção da solução em ambiente de produção para os parceiros Vodafone.
- Aperfeiçoamento de ideias iniciais.
- Requisitos não funcionais de tempo mais restritos, ou seja, os processos devem ser feitos o mais depressa possível; a partir de um certo limite a solução torna-se inaceitável.
- Necessidade de o método funcionar com várias versões do Windows (95,98,NT,2000,XP)

Este motivos originaram as seguintes mudanças:

- O “Componente Web” que comunica com a biblioteca tornou-se uma applet JAVA ou um controlo ActiveX, tendo a biblioteca que ser compatível com o dois.
- Tornou-se pouco aceitável a espera para o descarregamento da JAVA Virtual Machine. O uso do controlo ActiveX tornou-se preferível.
- O número que identifica o computador continua a ser principalmente o endereço MAC, pelas razões indicadas nos requisitos; contudo, para máquinas que não possuam

adaptador de rede, será usado o número de série do volume. Esta alteração será transparente a alto nível, já que continua a ser um número atribuído a uma máquina; deixa-se a cargo do software a sua distinção.

- Para o caso de máquinas com várias placas de rede ficou decidido que seriam guardados todos os endereços MAC dessas placas e que qualquer um desses endereços serviria para aceder a uma aplicação que implemente esta solução.
- Devido ao facto da primeira versão do Windows 95 não implementar as funções de rede Winsock 2.0 (onde se inserem as funções SNMP usadas), teve de ser feita uma adaptação ao código Visual Basic que comunica com a biblioteca de acesso ao hardware; assim, o código identifica a versão do Windows onde é executado recorrendo a métodos fiáveis da Windows API (*Application Programming Interface*). A partir deste ponto, é chamada a biblioteca correspondente (syspar.dll ou syspar95.dll). A razão da separação deve-se ao facto de o controlo se recusar a executar em Windows 95, provavelmente por causa da biblioteca ter sido compilada com o Winsock 2.0. Com estas mudanças, e depois de discussão com o grupo de projecto, ficou decidido que no sistema operativo Windows 95 seria sempre tirado o *Volume Serial Number* do disco para evitar erros. Estes problemas foram identificados na realização dos testes unitários (ver Anexo C).

## Descrição da Implementação

A solução é composta por três componentes principais:

- Biblioteca de acesso ao hardware
- Applet JAVA de acesso à biblioteca
- Controlo de ActiveX de acesso à biblioteca

A utilização dos componentes de acesso à biblioteca é comum; apenas é necessário incluir um objecto numa página HTML (também pode ser feito a partir de qualquer linguagem executada no servidor, como PHP ou JSP) e usar uma linguagem de *scripting* (como JavaScript) para aceder a uma função definida dentro do objecto.

### Arquitectura Implementada

Este é o diagrama em camadas da arquitectura efectivamente implementada:

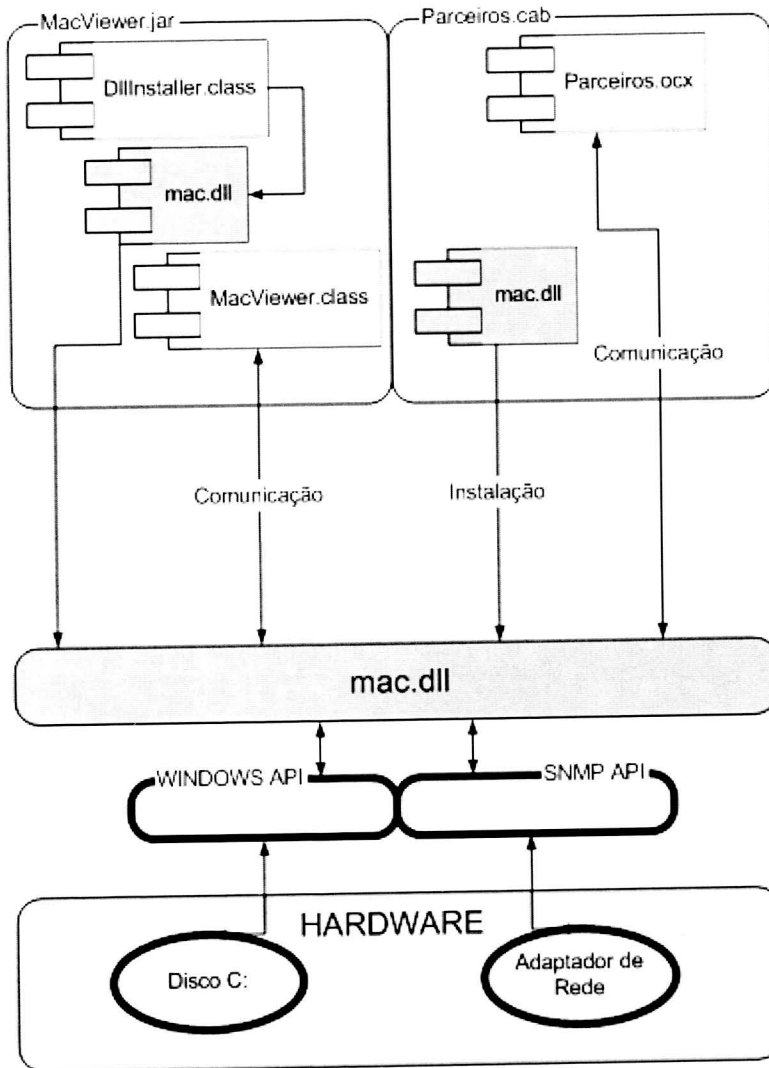


Fig. 4-3 Validação de Hardware - Arquitectura Implementada

### **Biblioteca de acesso ao hardware**

Esta biblioteca consiste numa DLL Win32 e encontra-se dividida em várias partes:

- Obtenção do endereço MAC, acedendo ao adaptador de rede por SNMP
- Obtenção do Volume Serial Number, acedendo ao disco local “C:”
- Ponto de entrada na DLL para o controlo ActiveX
- Ponto de entrada na DLL para a applet JAVA

Na figura 4-4 encontra-se o diagrama da arquitectura da implementação SNMP:

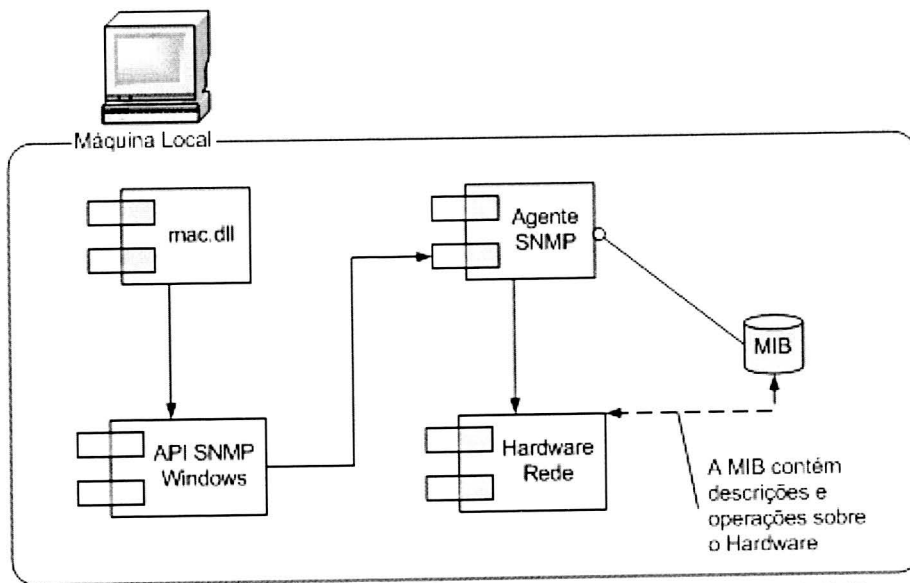


Fig. 4-4 Validação de Hardware - Arquitectura da biblioteca - SNMP

Neste caso não são usadas as potencialidades de gestão de redes remota do SNMP, já que tudo é feito localmente, tal como mostrado na figura acima.

Existe um acesso por parte da biblioteca *mac.dll* à biblioteca standard do Windows *snmp.dll*, que serve de interface de acesso à MIB do computador local.

O código desta biblioteca foi produzido com a ferramenta de desenvolvimento Microsoft Visual C++ 6.

### **Applets JAVA**

Este componente consiste na realidade em duas applets diferentes, uma que trata da instalação do DLL de acesso ao hardware e outra que comunica com essa DLL obtendo o endereço MAC ou Volume Serial Number.

As applets são as seguintes:

- `DllInstaller.class` – Applet de instalação
- `MacViewer.class` – Applet de acesso ao hardware



## Applet de Instalação

Esta applet consiste na classe `DllInstaller.class`, cujo fluxo se pode ver na figura 4-5:

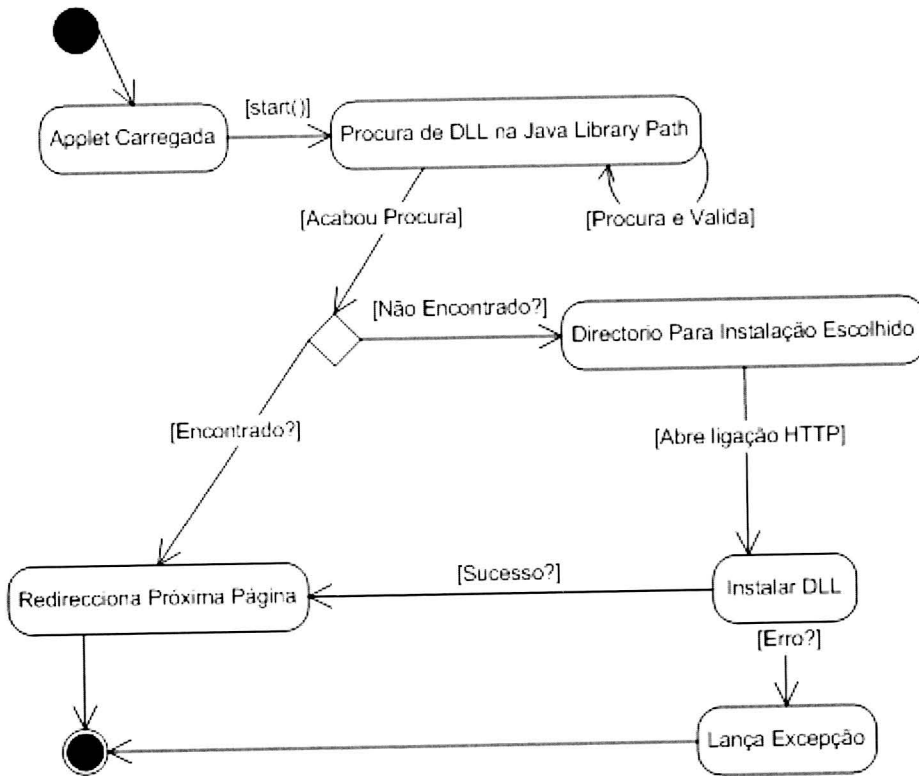


Fig. 4-5 Validação de Hardware - Diagrama de Estados da Applet de Instalação

O “Java Library Path” referida no diagrama é uma propriedade de sistema do JAVA que especifica os directórios aonde procurar as bibliotecas.

## Applet de acesso ao hardware

A segunda applet serve para o acesso ao hardware e precisa necessariamente da biblioteca instalada com a applet anterior.

Esta applet apenas carrega esta biblioteca em memória, cria um acessor para a função exportada pela biblioteca (de acordo com a especificação JNI) e executa-a.

## Autenticação do ficheiro .jar

Para algum deste código poder ser executado na máquina cliente é necessário autenticar o código, já que são usadas operações potencialmente perigosas, nomeadamente:

- Acesso a propriedades de sistema

```
System.getProperty("java.library.path");
```

- Acesso ao disco do cliente

```
f=new java.io.File(home + "/" +  
((java.util.jar.JarEntry)j.getJarEntry()).getName());
```

- Carregamento de bibliotecas

```
System.loadLibrary("mac");
```

Todas estas operações são proibidas numa applet não certificada, a não ser que explicitamente sejam dadas permissões a um certo domínio num ficheiro **java.policy**, o que tipicamente não acontece.

Usando a assinatura digital aparece um aviso na máquina cliente a avisar que o código é autenticado por uma certa entidade (sendo necessário também autenticar a entidade junto duma entidade certificadora).

Neste caso foram usadas assinaturas geradas para teste com o comando `makecert` do kit de autenticação fornecido com o JDK 1.3:

```
keytool -genkey -alias signFiles -keypass <password> -keystore <nome_da_keystore> -  
storepass <password_da_keystore>
```

Com este comando autentica-se o ficheiro .jar:

```
 jarsigner -keystore <nome_da_keystore> MacViewer.jar signFiles
```

## Controlo ActiveX

Este componente consiste num controlo ActiveX extremamente simples desenvolvido em Visual Basic 6 e distribuído num ficheiro .cab, sendo a sua instalação automática em Internet Explorer 4 ou superior.

As suas funções são:

1. Criar um wrapper da função `snmpMac()` definida na biblioteca de acesso ao hardware.
2. Executar essa função

### Parceiros.ocx

Este é o controlo propriamente dito, extremamente simples já que toda a lógica de instalação é tratada pelo seu contentor, o ficheiro `Parceiros.cab`.

### Parceiros.cab

O empacotamento do controlo foi feito usando uma ferramenta fornecida com o Visual Basic 6, o *Package and Deployment Wizard*, em que o processo de criação é o seguinte:

- Especificar o controlo a empacotar
- Especificar as dependências a serem distribuídas directamente (neste caso a DLL de acesso ao hardware)

- Criar o ficheiro .cab

Depois deste processo foi necessário editar o ficheiro .inf (guia de instalação) criado por omissão dentro do .cab para lhe acrescentar lógica adicional, na secção [mac.dll]

- RegisterServer=yes – Permite que o DLL seja registado no Windows
- DestDir=10 – Faz com que o DLL seja instalado no directório do Windows do sistema.

Com estas informações adicionadas ao ficheiro .inf é necessário então um novo empacotamento em ficheiro .cab, usando, p.ex. a ferramenta da Microsoft `cabarc.exe`

### Autenticação do ficheiro .cab

Devido ao acesso que faz ao hardware e à instalação dessa mesma biblioteca na máquina cliente, tornou-se necessário autenticar o ficheiro.

Para além disso, a configuração base do Microsoft Internet Explorer (*Medium Security*) simplesmente ignora os controlos ActiveX que não sejam certificados, por motivos de segurança.

Foram usadas as ferramentas da Microsoft disponíveis em

<http://msdn.microsoft.com/MSDN-FILES/027/000/219/codesign.exe>

Para gerar um certificado *self-signed*, tipicamente usado para testes, procede-se da seguinte forma

```
D:\inetsdk\bin>MakeCert -sk myNewRootKey -r -n "CN=Miguel Pereira" -ss myNewRoot
saveCertificate.cer
```

Para usar este certificado, que se encontra agora armazenado na *store* “myNewRoot”, basta executar o comando:

```
D:\inetsdk\bin>signcode -s myNewRoot <ficheiro_a_assinar>
```

Deste modo o ficheiro .cab fica autenticado e, aquando da sua execução no Internet Explorer, aparece um aviso a perguntar se aceita correr um controlo ActiveX assinado com aquele certificado.

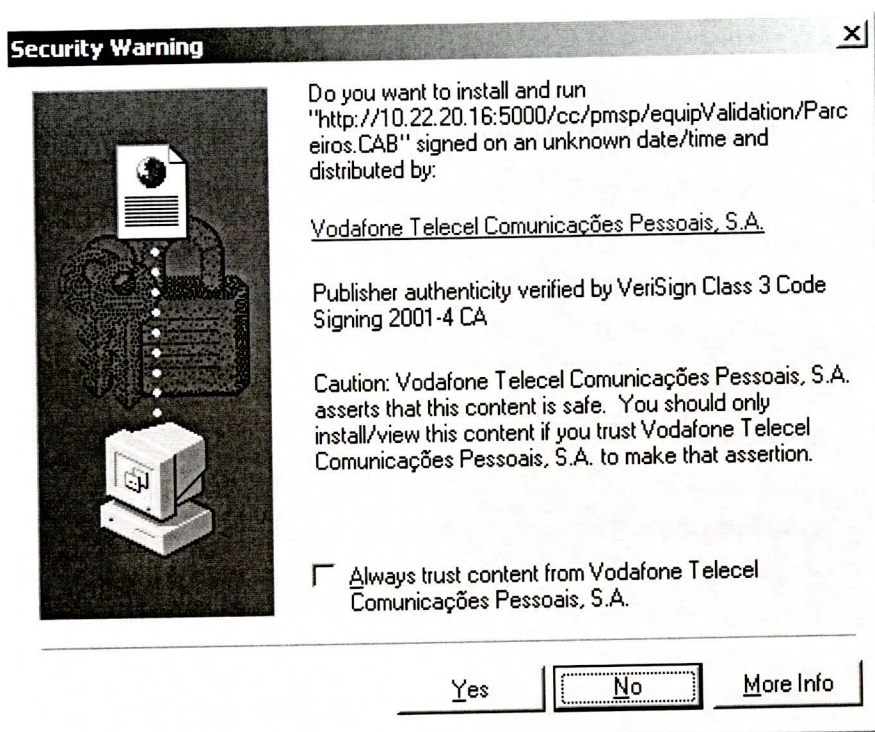


Fig. 4-6 Validação de Hardware - Aviso do Internet Explorer

Numa fase posterior foi adquirida uma assinatura (em [www.verisign.com](http://www.verisign.com)) que certifica a Vodafone Telecel S.A como de confiança para assinar código passível de distribuição em ambiente Web (ver figura 4-6)

## 4.2 Validação do *hardware* cliente na aplicação WebParceiros

A integração da solução descrita em 4.1 na aplicação WebParceiros foi uma ramificação ao projecto “Portal de Serviços”, algo que não estava planeado inicialmente. Todavia, dado o esforço de desenvolvimento gasto e a importância da adopção da solução foi decidido introduzir este capítulo no relatório.

A aplicação WebParceiros encontra-se descrita em mais pormenor na secção 3.3.; a presente secção apenas se dedica a apresentar as opções novas desenvolvidas no âmbito desta ramificação.

Esta integração nasceu de uma reunião em que a equipa foi informada da sua necessidade. Para melhor explicitar as decisões apresenta-se a seguir aqui um excerto da acta da reunião ocorrida em 2002/07/01 e que especifica os Requisitos Iniciais.

### Requisitos Iniciais

<Extracto da Acta contida no documento Acta de Reunião \_20020701.doc>



Foi comunicado pelo Eng. Henrique Necho que a direcção do MIS decidiu em reunião de directores que se implementaria em ambiente de produção a solução “Validação do Hardware Cliente” na aplicação Web Parceiros.

A data expectável de entrada em produção é do dia 28 de Julho, dia de lançamento de uma nova *release* da aplicação. Para este efeito é necessário entregar a aplicação para System Test no dia 14 de Julho.

Os problemas que foram identificados ( e sua resolução ) foram os seguintes:

### ***Como identificar máquinas sem placa de rede?***

Existem várias soluções, mas aquela que apresenta o melhor compromisso entre fiabilidade e exequibilidade é a do Volume Serial Number do disco. Apesar deste número mudar a cada formatação, o Eng. Henrique Necho disse que isso não era um caso típico numa loja. Em Base de Dados fica, então, um serial number que no caso de placa de rede é o endereço MAC e no caso de não existir placa de rede é o Volume Serial Number do disco.

### ***Master User é ou não identificado pelo serial number?***

Esta questão deu azo a muitas opiniões, mas foi resolvida por um telefonema ao Eng. Artur Baptista, que esclareceu que se vai confiar plenamente no Master User, tendo ele a possibilidade de activar qualquer máquina sem ser ele próprio identificado pelo serial number.

Uma solução ainda mais segura implicaria a necessidade de activação da própria máquina do Master User, mas em termos logísticos isso seria custoso porque precisaria de pedidos de acesso, cartas ou um técnico a visitar cada agente.

### ***Onde aparece a opção de activação no Web Parceiros?***

Foram discutidas 2 hipóteses principais:

1. Uma página intermédia que apareceria depois de se carregar no botão “Activar” da página principal do Web Parceiros. Essa página consistiria nas opções de “Activar o utilizador” e “Activar PC”.
2. As opções ficariam no menu principal do Web Parceiros, já depois de *login* e só visíveis para Master Users.

A hipótese escolhida foi a 2.

### ***Como fazer para reduzir ao mínimo os tempos de activação de uma máquina?***

Foi calculado que para um agente com ligação RDIS 64k (a mais baixa definida nos requisitos mínimos do WebParceiros para os agentes) e sem máquina virtual Java, a instalação poderia demorar mais de 10-15 minutos.

Como esta activação é tipicamente feita em mais do que um PC por loja considerou-se este tempo um pouco exagerado.

O Miguel Pereira sugeriu que fosse tentada uma aproximação de instalação remota por ActiveX (Microsoft), que reduziria o tempo a algo bastante menor, pelo menos se a versão do Windows fosse recente.

Foram discutidas várias soluções de implementação e trade-offs entre segurança e exequibilidade da solução. Ficou definido que os requisitos principais seriam os seguintes:

1. Master User de um Agente pode activar, listar e remover activação de máquinas de cada uma das suas lojas
2. A lista de serials está associada a um site (loja), que por sua vez está associado a um agente
3. Seria interessante poder identificar a loja onde o utilizador faz *login*, aparecendo na aplicação algo de género “Bom dia senhor M\_\*\*\*\*, encontra-se na loja X”

## Análise

Dos requisitos especificados foram tirados dois casos de uso, o da activação de uma máquina pelo MASTER USER e o de um *login* por um NORMAL\_USER.

## Diagrama de Casos de Uso

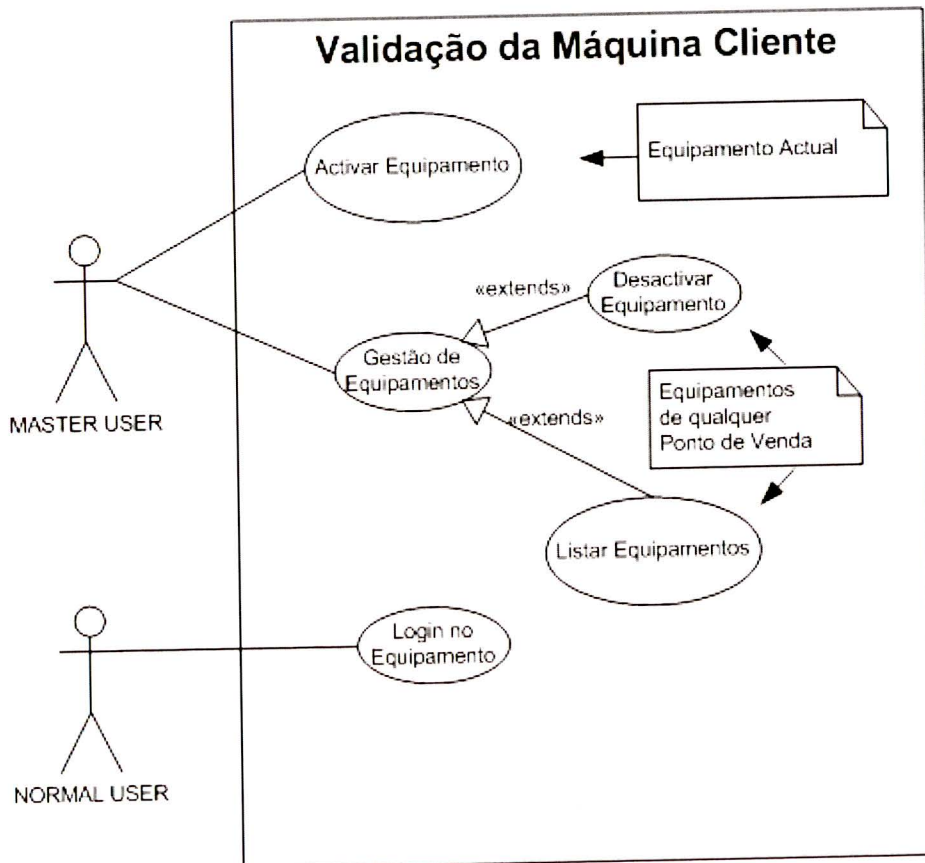


Fig. 4-7 Validação de Hardware no WebParceiros - Casos de Uso

Diagramas de sequência

Login com máquina autenticada

Neste diagrama é ilustrado o processo de login bem sucedido com autenticação tanto de user/pass como de máquina.

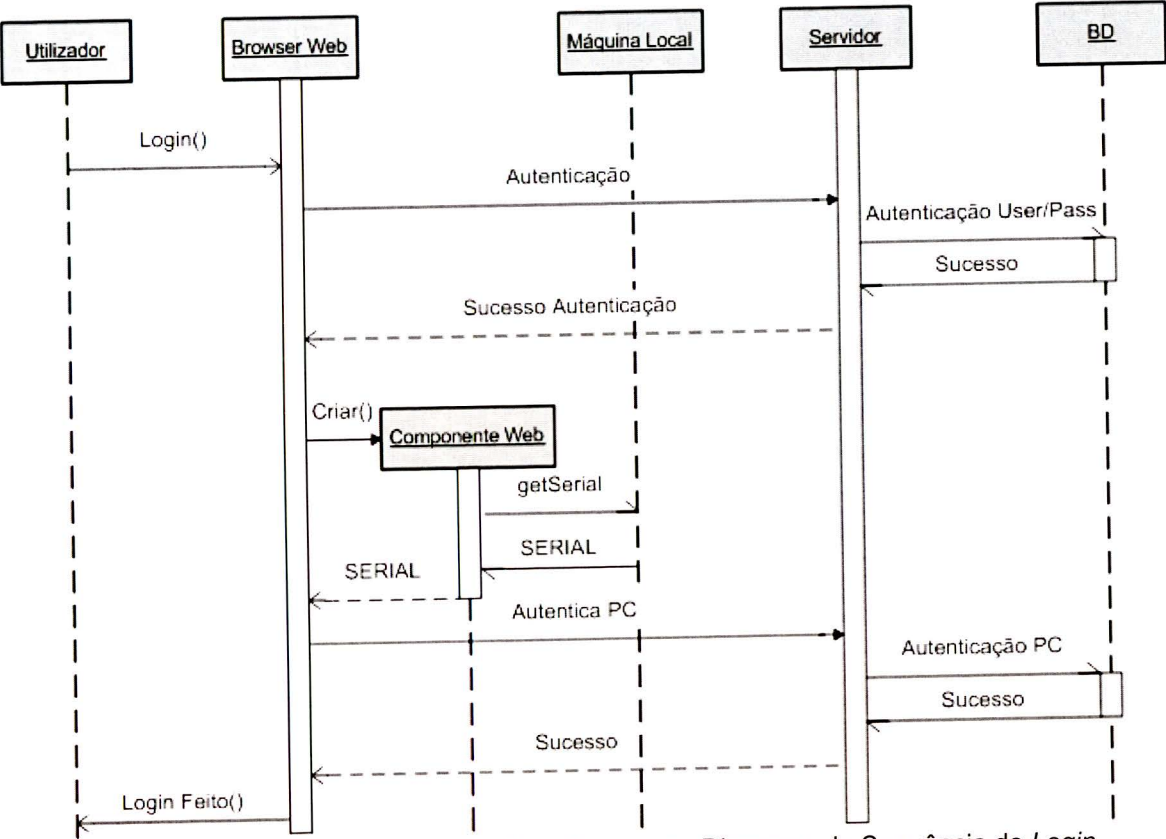


Fig. 4-8 Validação de Hardware no WebParceiros - Diagrama de Sequência do Login



## Activação de máquina

Neste diagrama é ilustrado o processo da activação de uma máquina.

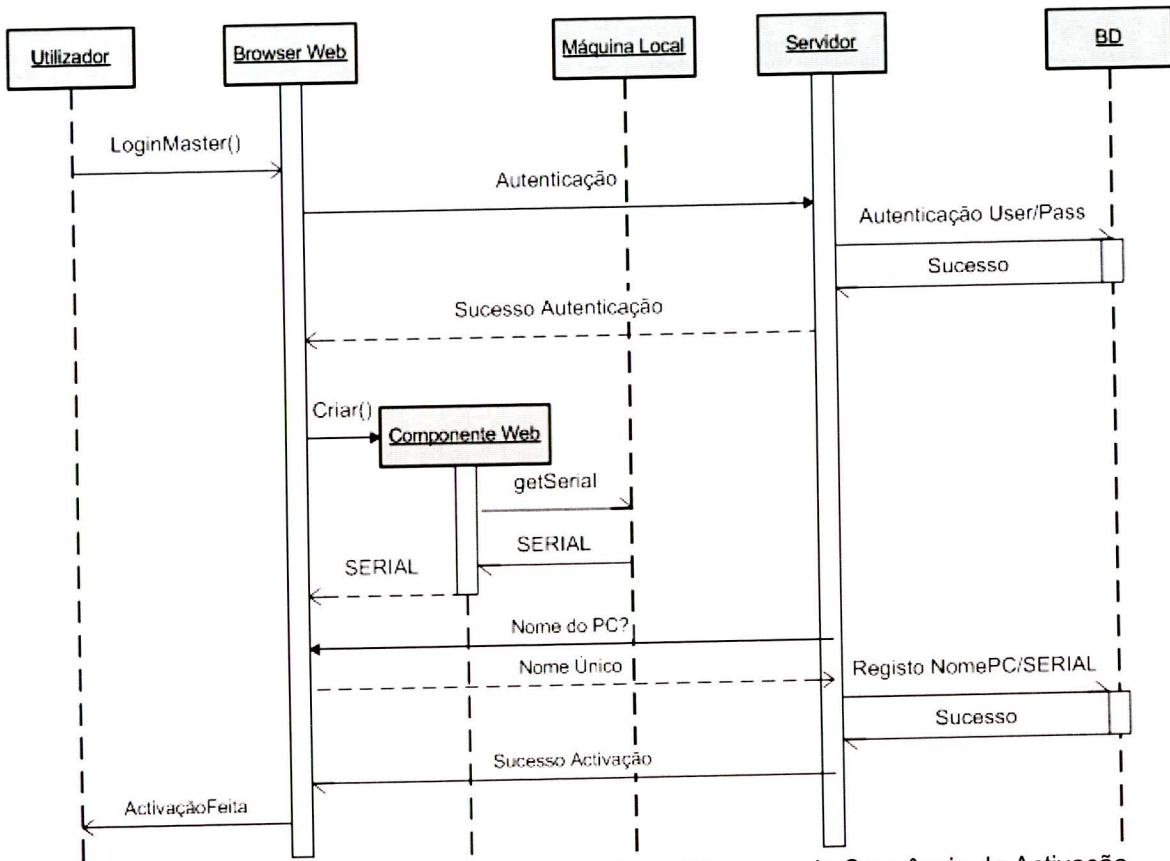


Fig. 4-9 Validação de Hardware no WebParceiros - Diagrama de Sequência da Activação

### Acesso WebParceiros C/ Validação Equipamento

A figura 4-10 apresenta o diagrama final do fluxo de *login* para um utilizador Normal.

O ficheiro `SSPPMDistpatcher.java` era o único servlet que tratava do *login* inicialmente. Com estas extensões, foi desenvolvido por outro elemento da equipa de projecto o servlet `SSPMEquipValidation.java`, que trata de autenticar o utilizador

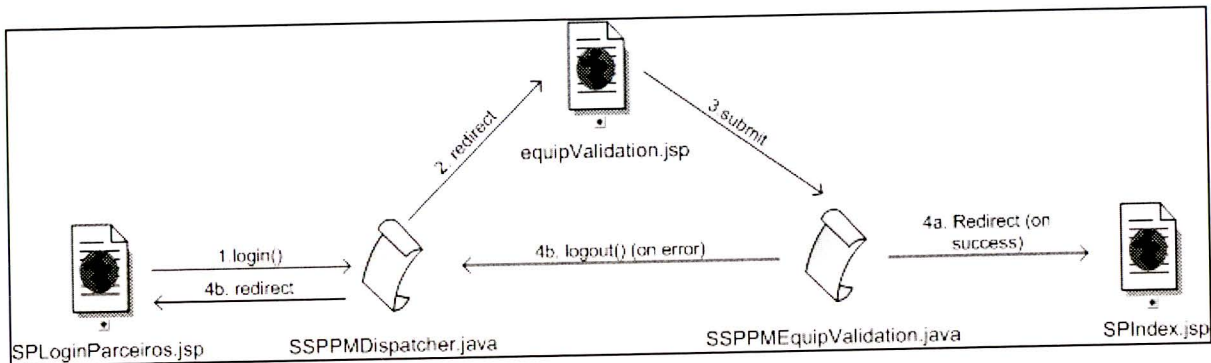


Fig. 4-10 Validação de Hardware no WebParceiros - Fluxo de acesso de um utilizador NORMAL

A aplicação WebParceiros, irá ser alterada em dois pontos: No *login* dos Normal Users e com a inclusão de duas novas opções para o Master Users.

O *login* do Normal User será alterado; para além do *login*, *password* e estado serem validados positivamente, o *serial\_number* do equipamento que está a ser utilizado tem que constar na lista de *serial\_number* válidos, que são adicionados sempre que o Master User regista um equipamento. Para que seja implementada esta funcionalidade terá que ser alterada a servlet `SSPPMDistpatcher.java`.

Para que seja efectuado o registo e gestão dos equipamentos irá ser incluído um novo grupo (Equipamento Informático) com duas novas opções (Registar Equipamento; Gerir Equipamento), explicação pormenorizada abaixo.

### Protótipo Gráfico Desenvolvido

Pode ser encontrado no Anexo A.2 - Protótipo Gráfico para a integração de novas opções no WebParceiros

Descrição da Implementação

Alteração Modelo de Dados

Com a inclusão desta nova opção irão ser necessárias alterações ao modelo de dados, nomeadamente a criação de uma nova tabela para guardar a informação de cada equipamento e a sua ligação à tabela SP\_SALES\_POINT. Na figura 4-11 está representada a inclusão da nova tabela

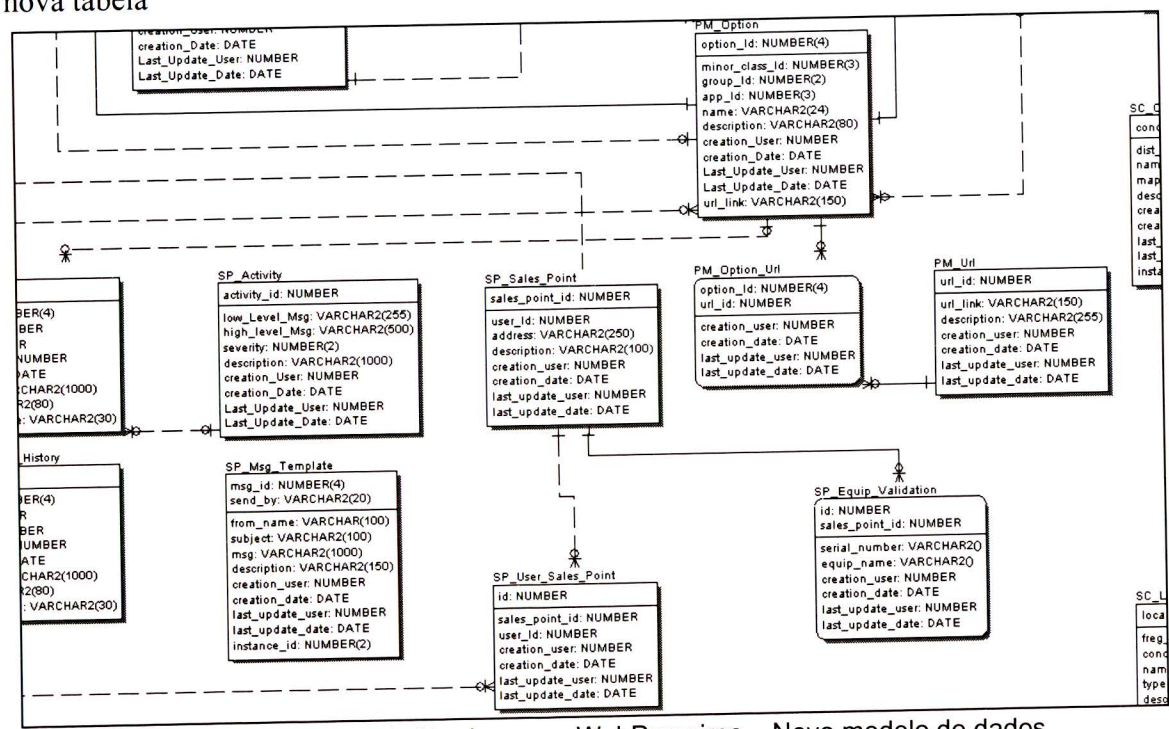


Fig. 4-11 Validação de Hardware no WebParceiros – Novo modelo de dados

- Tabela: SP\_EQUIP\_VALIDATION:
- Sequência: SP\_EQUIP\_SEQ
- Trigger: SP\_EQUIP\_VALIDATION\_TRIGGER
- Chave Estrangeira: FK\_SP\_EQUIP\_VALIDATION
- Chave Primária: PK\_SP\_EQUIP\_VALIDATION



### ***PMSPEquipValidationEJB (EJB de suporte)***

Para implementar toda a lógica de negócio necessária às operações foi desenvolvido um EJB, solução mais limpa até porque foi necessário editar (incluindo desenho das páginas) vários ficheiros já existentes e desenvolvidos por outras pessoas. Assim, ficou decidido que o EJB forneceria métodos que depois seriam acedidos a partir das páginas JSP.

Os métodos implementados neste EJB foram os seguintes:

```
public Vector list(PMSPUserInfo userInfo, String salesPointId) throws PMSPEException
```

Lista os nomes dos equipamentos associados a um certo ponto de venda. Se o argumento `salesPointId` for nulo, lista todos os equipamentos associados a um determinado agente.

```
public void update(PMSPUserInfo userInfo, String newEquipName, String equipName,
String oldSalesPointId, String newSalesPointId) throws PMSPEException
```

Actualiza o nome do equipamento de determinado `oldSalesPointId` para `newEquipName`. O argumento `newSalesPointId` não é neste momento utilizado.

```
public boolean checkPC(PMSPUserInfo userInfo, String equipName, String
salesPointId) throws PMSPEException
```

Verifica se o equipamento de nome `equipName` existe na loja com id `salesPointId`.

```
public boolean checkSerial(Vector serial) throws PMSPEException
```

Verifica se algum dos números do vector de serial numbers `serial` já existe.

```
public Vector checkSerialData(Vector serial) throws PMSPEException
```

Retorna um vector contendo o endereço da loja e o nome do equipamento, se existir.

```
public PMSPEquipInfo validate(PMSPUserInfo userInfo, Vector serial) throws
PMSPEException
```

Este método é usado na sequência de *login* de um utilizador normal. Valida o(s) números presentes no vector `serial`. Se válidos, retorna um objecto `PMSPEquipInfo` (classe desenvolvida pelo grupo de projecto) que contém informação acerca do equipamento e é que posto em sessão no caso do *login* ser bem sucedido.

```
public void activate(PMSPUserInfo userInfo, int salesPointId, Vector serialNums,
String pcName) throws PMSPEException{
```

Este método é usado na activação de um equipamento por parte do MASTER USER.

```
public void deactivate(String pcName, String salesPointId) throws PMSPEException
```

Este método desactiva todos os números associados ao equipamento com o nome `pcName` na loja com id `salesPointId`.

```
public void deactivate(Vector serialNums) throws PMSPEException
```

Este método desactiva um equipamento baseado no(s) seu(s) serial number(s).

## 5 Conclusões

### 5.1 Considerações Pessoais

Uma das coisas mais importantes foi o facto de me ter sentido parte integrante desta empresa, com motivação para ajudar na sua construção e constante melhoramento.

Devo acrescentar que o facto de ter trabalhado com o meu colega Joel Alvim proporcionou uma experiência valiosa de trabalho em grupo, completando a adquirida no passado nos trabalhos académicos e realçando a importância da diversidade de opiniões na construção de melhores soluções e melhores processos.

As pessoas com quem trabalhei de perto na Vodafone Telecel S.A ensinaram-me muito a muitos níveis; nível empresarial para uma melhor visão do funcionamento de uma organização desta dimensão; nível humano porque me mostraram como aplicar os conhecimentos adquiridos de maneira a privilegiar o utilizador final e a valorizar as relações inter-pessoais numa empresa; nível tecnológico porque tive a oportunidade de aumentar os meus conhecimentos de engenharia informática em toda a sua abrangência.

### 5.2 Contribuições

Este trabalho desenvolvido no âmbito do estágio curricular foi importante para a empresa Vodafone Telecel S.A. pelas seguintes razões: permitiu avaliar o produto iPlanet Portal Server 3.0 e com o piloto de testes realizado foram estudadas as mais-valias que podem decorrer do uso deste produto, nomeadamente as vantagens de uma centralização da autenticação tanto a nível de acesso interno como externo. Além disto, com a ramificação do projecto, foi encontrada uma solução de segurança fiável e económica para a identificação única de um cliente que aceda a um sítio Web; esta solução encontra-se neste momento implementada e em uso por parte dos agentes autorizados Vodafone.

### 5.3 Trabalho Futuro

Apesar de este estágio, na sua vertente curricular, terminar com a entrega e apresentação do relatório, o prazo foi alargado por parte da direcção de sistema de informação da Vodafone Telecel S.A para validar a plataforma e ultimar o projecto. A decisão final sobre a continuidade do projecto está dependente destas ocorrências, sendo portanto adiada para além do estágio curricular.

Em termos de melhoramentos existe a necessidade de criar aplicações de gestão de utilizadores, algo que irá ser feito a curto prazo.

No futuro esta plataforma poderá tornar-se num verdadeiro portal empresarial, que agregue e concentre informação e aplicações úteis para os parceiros Vodafone.



## Referências e Bibliografia

**Documentos produzidos pelos elementos do grupo de projecto “Portal de Serviços”, Miguel de Castro Pereira e Joel Rebelo de Castro Alvim:**

admin\_guide\_20020306 - Revisão de “iPlanet Portal Server 3.0 Administration Guide”,  
Fonte: <http://docs.iplanet.com/>, 2001

index\_iportal\_arch\_20020304 – Revisão do documento “iPlanet. Portal Server 3.0 Overall Architecture”, Fonte: <http://docs.iplanet.com>, 2001

ldap\_overview\_20020313 - Visão geral do LDAP, Fontes: <http://www.netscape.com>,  
<http://www.openldap.org>, 2002

weblogic\_intro\_20020313 – Revisão de “Introduction to WebLogic Server 5.1”, Fonte:  
<http://edocs.bea.com>, 2001

wp\_gateway\_20020315 – Revisão do *White Paper* da componente Gateway do iPlanet Portal Server, Fonte: <http://developer.iplanet.com>, 2001

### Documentação de outros projectos internos:

GD\_WebParceiros – General Design do projecto “Solução Web para Parceiros”, Miguel Pereira e Benjamin Ferreira, 2001

### Sites Web consultados

VODAFONE TELECEL, S.A. – Páginas do site <http://www.vodafone.pt>, Vodafone Telecel S.A., 2002

BEA SYSTEMS – Páginas do site <http://www.bea.com>, Bea Systems, 2002

IPLANET – Páginas do site <http://www.iplanet.com>, iPlanet, 2002

MICROSOFT – Páginas do site <http://msdn.microsoft.com>, Microsoft, 2002

SUN MICROSYSTEMS – Páginas do site <http://java.sun.com>, Sun Microsystems, 2002

BORLAND – “Three Ways To Get Your System’s Network MAC Address”, Borland Developer Network (<http://community.borland.com/article/0,1410,26040,00.html>), Borland, 2002

JAVAWORLD – Páginas do site <http://www.javaworld.com/>, JavaWorld, 2002

## ANEXO A – Detalhes de Implementação

Neste anexo são apresentados os detalhes de implementação de três sub-projectos:

- Aplicação “Feedback Reports”
- Protótipo Gráfico para a Integração de Novas Opções no WebParceiros
- Sítio Web Interno de Projecto

### A.1 Aplicação Feedback Reports

#### Motivação e Requisitos

A aplicação FeedbackReports foi uma ideia do Eng. Henrique Necho que teria três objectivos principais:

- Adicionar mais uma aplicação ao portal, de modo a mostrar a sua escalabilidade.
- Proporcionar aos operadores que recebem chamadas relativas às aplicações WebParceiros e SFAWeb um modo de armazenarem características desse pedido.
- Fornecer estatísticas relativas a essas chamadas.

Os dois últimos pontos foram considerados os requisitos iniciais da aplicação a desenvolver.

#### Arquitectura

A arquitectura da aplicação foi definida como sendo de três camadas; a camada do cliente (PC), a camada de apresentação e a camada de negócio, como se pode ver na figura seguinte:

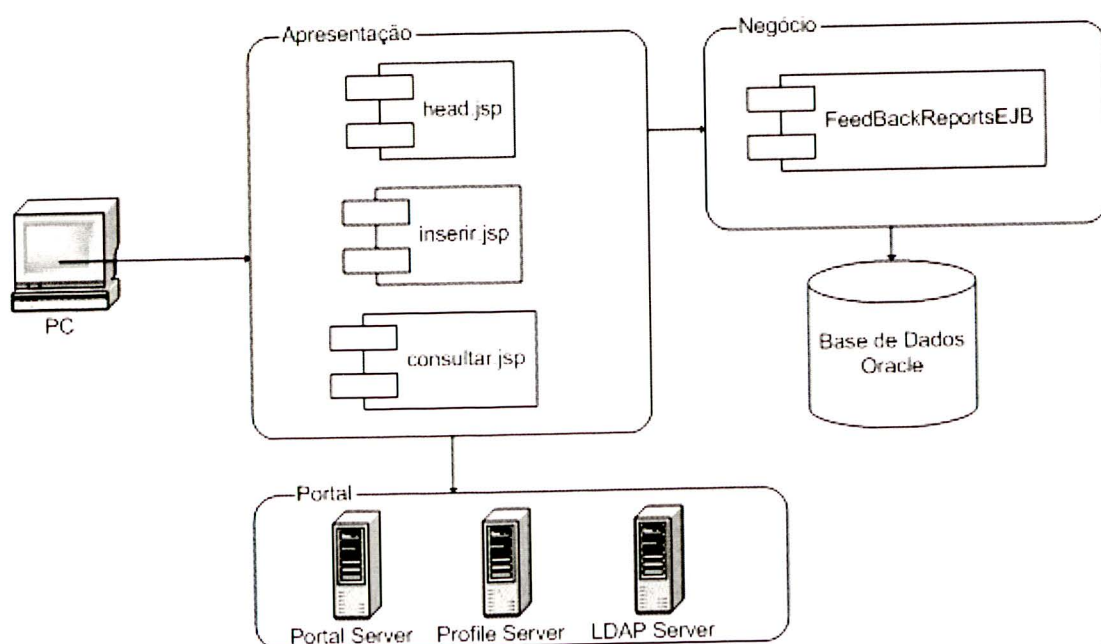


Fig. A-1 Feedback Reports - Arquitectura



A camada cliente trataria da apresentação do conteúdo final ao utilizador, a camada de apresentação conteria código para gerar esse conteúdo e a camada de negócio serviria de interface de acesso à base de dados onde iriam ser armazenados os dados.

Análise

Diagrama de Classes

Nesta fase foi desenhado o esquema de classes que iria dar origem ao modelo relacional a implementar na base de dados. Esse esquema é apresentado a seguir

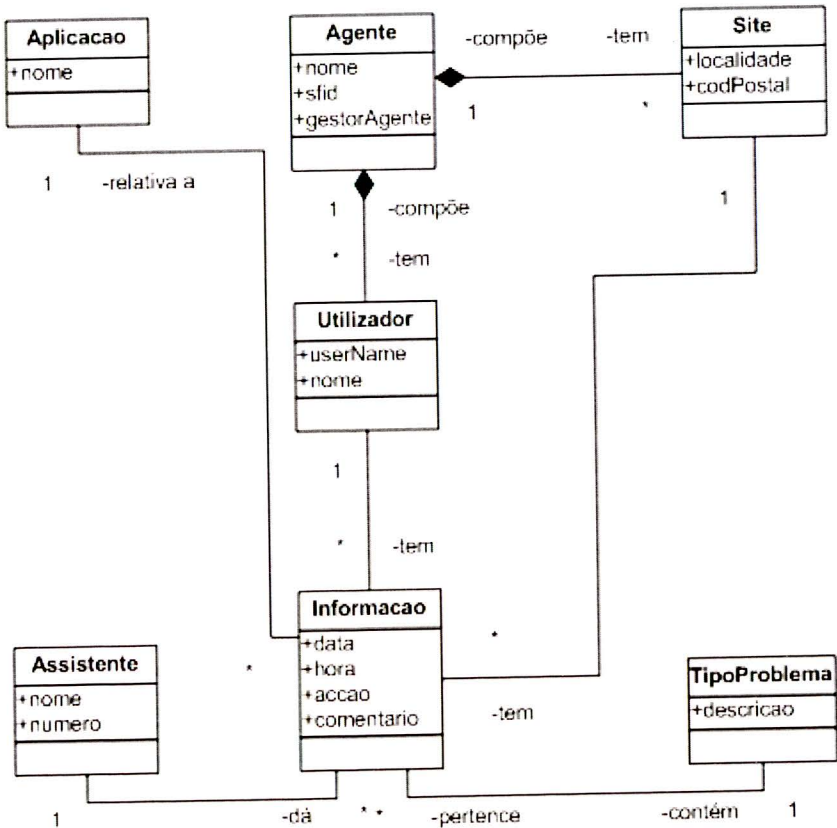


Fig. A-2 Feedback Reports - Diagrama de Classes

### Diagrama de Casos de Uso

O diagrama de casos de uso, extremamente simples, foi idealizado e construído

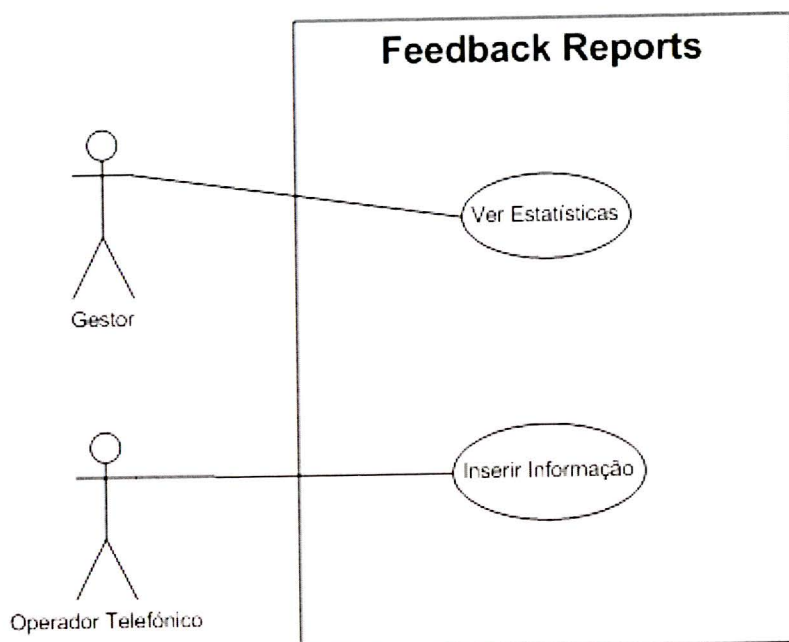


Fig. A-3 Feedback Reports Diagrama de Casos de Uso

Nesta fase foram também idealizadas as páginas HTML para cada um dos casos de uso.

### Descrição da Implementação

#### Interface

A interface da aplicação foi implementada recorrendo a HTML com uso de CSS (*Cascading Style Sheets*), de modo a tornar o aspecto facilmente configurável. O aspecto final é o seguinte:

[\[ Submeter Relatório \]](#) [\[ Estatísticas \]](#) [\[ Logout \]](#)



|                  |                          |
|------------------|--------------------------|
| Agente           | <div>Stand Asla</div>    |
| Localização      | <div>Amadora 2720</div>  |
| Aplicação        | <div>WebParceiros</div>  |
| User             | <div>N_FABREU</div>      |
| Tipo de Problema | <div>Conectividade</div> |
| Questão          | <div></div>              |
| Ação             | <div></div>              |

Inserir

Fig. A-4 Feedback Reports - Página de Inserção de Feedback

As caixas de texto “Localização” e “User” são dependentes da caixa “Agente”, ou seja, ao seleccionar um agente as caixas relativas à loja e ao utilizador são preenchidas automaticamente, sem haver novo carregamento da página (recorrendo ao Dynamic HTML).

[\[ Submeter Relatório \]](#) [\[ Estatísticas \]](#) [\[ Logout \]](#)



|                                                                                                 |                                                                                                              |
|-------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------|
| <div>Pesquisa de registos por agente</div> <div><div>Stand Asla</div><div>Pesquisar</div></div> | <div>Pesquisa de registos por utilizador</div> <div><div>M_AMARQUES</div><div>Pesquisar</div></div>          |
| <div>Pesquisa de registos por aplicação</div> <div><div>SFAWeb</div><div>Pesquisar</div></div>  | <div>Pesquisa de registos por tipo de problema</div> <div><div>Conectividade</div><div>Pesquisar</div></div> |
| <div>Top 5 dos Problemas</div> <div>Pesquisar</div>                                             |                                                                                                              |

Fig. A-5 Feedback Reports Página de Visualização de Estatísticas

### FeedbackReportsEJB

Este EJB apenas serve de invólucro (*wrapper*) aos métodos JDBC de acesso à base de dados. Assim, proporciona os métodos `select()`, `update()` e `insert()`

## A.2 Protótipo Gráfico para a integração de novas opções no WebParceiros

### Opções Master User

Na figura seguinte é apresentado o novo menu com as novas opções:

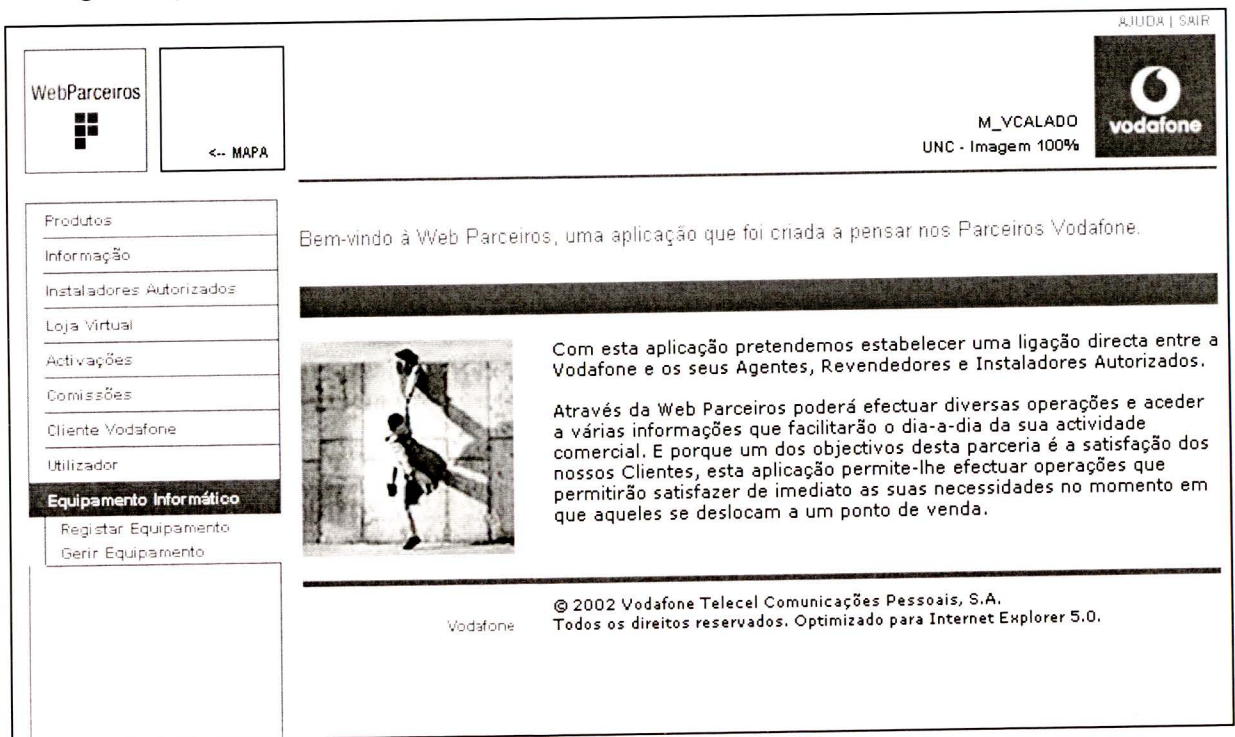


Fig. A-6 Novas Opções WebParceiros - Menu com novo grupo de opções



Opção Registrar Equipamento

Esta opção implementa a operação de Activação de uma Máquina e tem 3 passos, a saber:

- escolher
- confirmar
- resultado

Na figura seguinte é apresentada o ecrã correspondente ao passo “Escolher”, em que se introduz um nome para o equipamento e se associa a um determinado ponto de venda.

WebParceiros



<- MAPA

Produtos

Informação

Instaladores Autorizados

Loja Virtual

Ativações

Comissões

Cliente Vodafone

Utilizador

Equipamento Informático

Registrar Equipamento

Gerir Equipamento

AJUDA | SAIR

M\_VCALADO

UNC - Imagem 100%



escolher > confirmar > resultado >

DÚVIDAS

Registrar Equipamento

> Deve inserir o nome pelo qual deve pertence identificar este equipamento e indicar em que loja se encontra.

Nome Equipamento \*

( nome pelo qual pretende identificar este equipamento )

Ponto de Venda \*

C.Comercial Jumbo Alfragide, lj.10 - 2700 Amadora

( ponto de venda associado ao equipamento )

Os campos indicados com \* são de preenchimento obrigatório.

AVANÇAR >>

Fig. A-7 Novas Opções WebParceiros - Passo “Escolher” do registo

A seguir é apresentado o passo de confirmação de um registo:

WebParceiros



<- MAPA

Produtos

Informação

Instaladores Autorizados

Loja Virtual

Ativações

Comissões

Cliente Vodafone

Utilizador

Equipamento Informático

Registrar Equipamento

Gerir Equipamento

AJUDA | SAIR

M\_VCALADO

UNC - Imagem 100%



escolher > confirmar > resultado >

DÚVIDAS

Registrar Equipamento

> Confirma a associação entre o equipamento e a loja ?

Nome equipamento

Posto de Trabalho - Nº 1

Ponto de Venda

C.Comercial Jumbo Alfragide, lj.10 - 2700 - Amadora

<< ANTERIOR AVANÇAR >>

Fig. A-8 Novas Opções WebParceiros - Passo “Confirmar” do registo

Na figura seguinte é mostrada a página de resultado do registo, neste caso executado com sucesso:

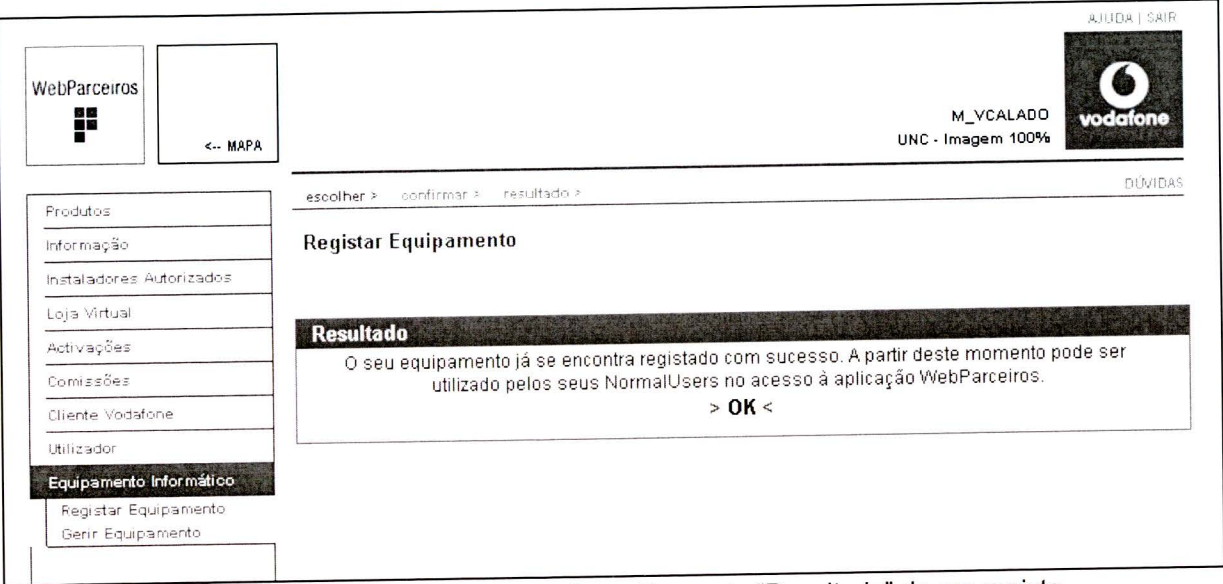


Fig. A-9 Novas Opções WebParceiros - Passo de “Resultado” de um registo

**Opção Gerir Equipamento**

Com esta opção pretende-se que o utilizador MASTER possa gerir os equipamentos registados em todas as suas lojas, entendendo-se por gerir a operação de “listar” e de “desactivar”

Na figura seguinte é mostrada a opção “Escolher” da opção de gestão de equipamento, em que o utilizador escolhe um dos seus pontos de venda

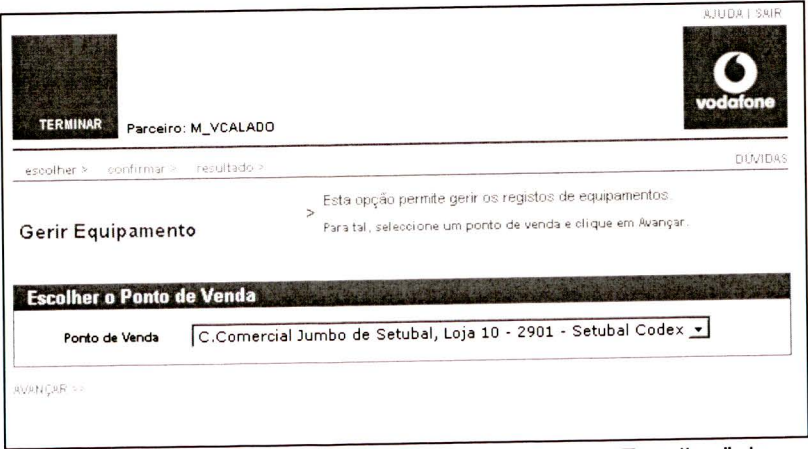


Fig. A-10 Novas Opções WebParceiros - Passo “Escolher” da gestão de equipamento



A figura seguinte ilustra o passo seguinte, em que é apresentada ao utilizador uma lista de equipamentos registados no ponto de venda que escolheu e em que pode editar o nome do equipamento ou desactivá-lo:

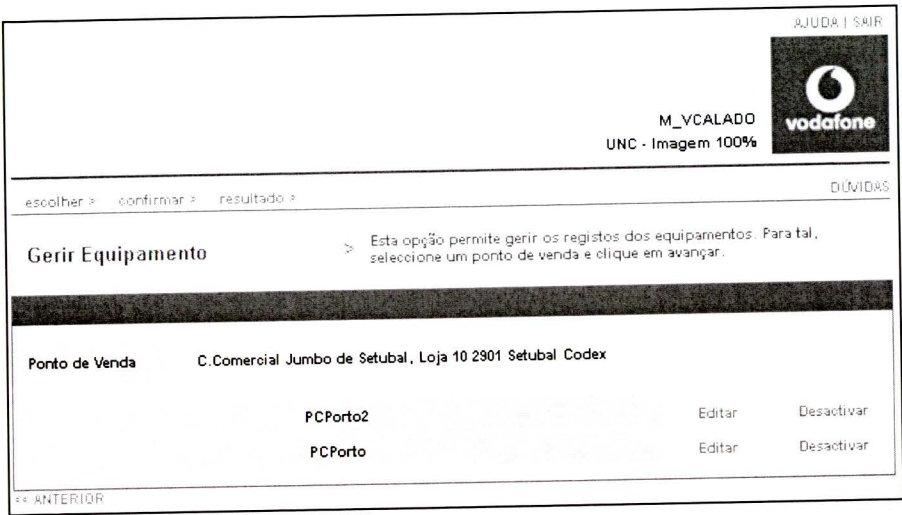


Fig. A-11 Novas Opções WebParceiros - Operação de listagem da gestão de equipamento

Quando a operação escolhida é Editar, a mesma página é mostrada, mas com a possibilidade de edição do nome do equipamento seleccionado, como se pode verificar na figura seguinte.

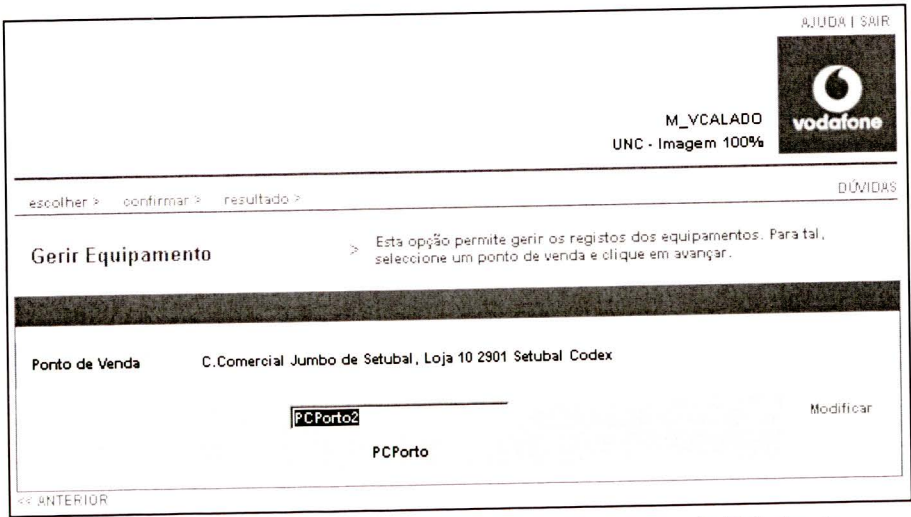


Fig. A-12 Novas Opções WebParceiros - Operação de Edição da gestão de equipamentos

No caso da desactivação de um equipamento, o sistema redireccionará para a página de confirmação.

A página de confirmação será diferente de acordo com a operação escolhida. O *design* está representado nas figuras A-13 e A-14.

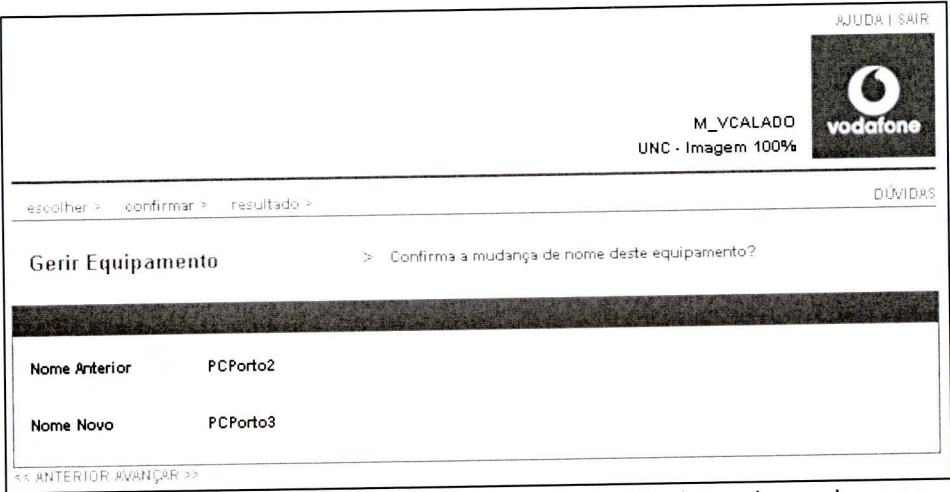


Fig. A-13 Novas Opções WebParceiros - Confirmação da mudança de nome

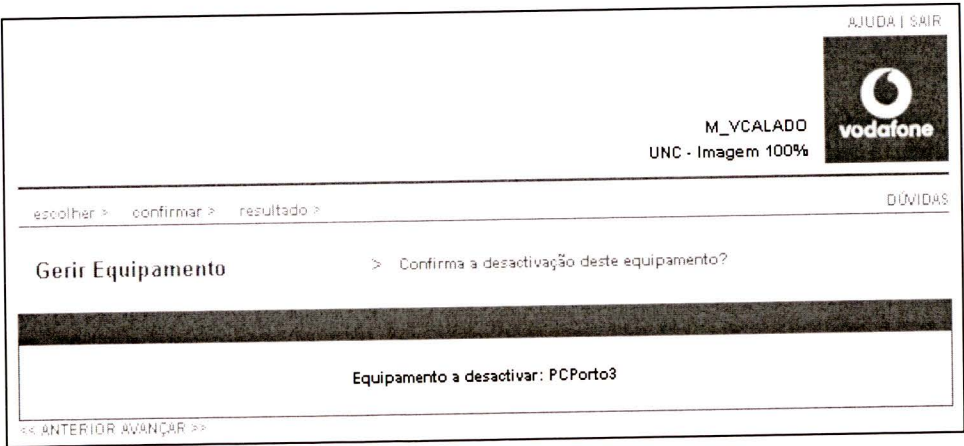


Fig. A-14 Novas Opções WebParceiros - Confirmação da desactivação

A página de resultado é apresentada a seguir:

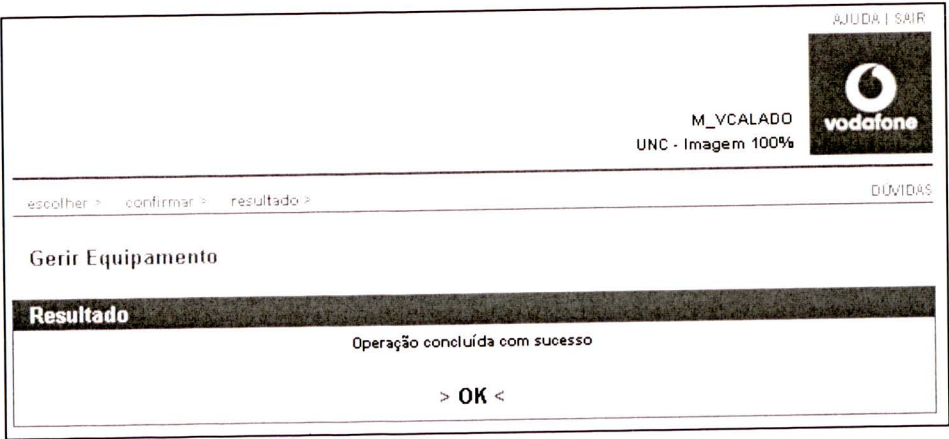


Fig. A-15 Novas Opções WebParceiros - Resultado da Operação

## A.3 Sítio Web Interno de Projecto

### Requisitos e Análise

A construção deste sítio Web, no início do projecto, preencheu um requisito fundamental: dar visibilidade ao projecto dentro da empresa e disponibilizar a documentação criada de modo a que todos os elementos da equipa lhe tivessem acesso.

A estrutura do sítio é simples e tem as seguintes secções:

- Breve Introdução
- Documentação

A primeira consiste num diagrama de arquitectura do projecto e de uma pequena introdução escrita.

A segunda opção compreende mais funcionalidades, como seja a consulta e a inserção de documentação. Esta documentação está estruturada por grupos, que são especificados aquando da sua inserção. A apresentação no ecrã é gerada por código JSP acedendo a um EJB para os métodos de suporte.

O diagrama de casos de uso seguinte sintetiza estas funcionalidades.

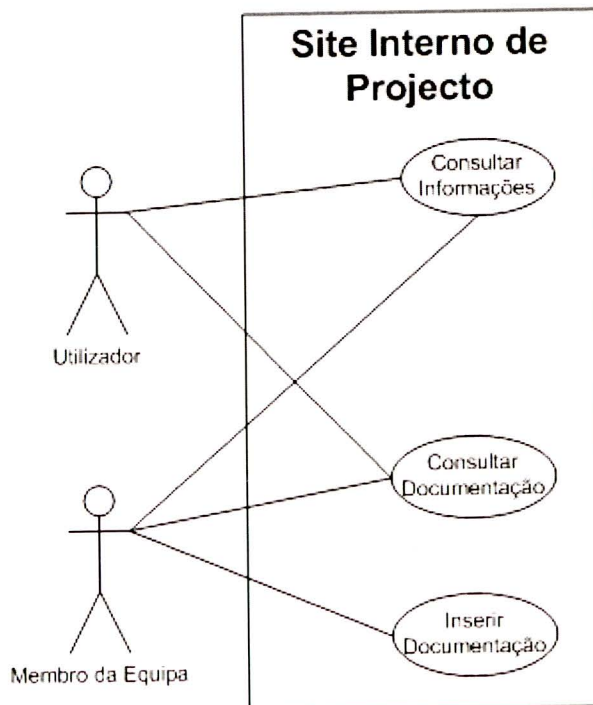


Fig. A-16 Sítio Web Interno de Projecto -  
Diagrama de Casos de Uso

Para suportar estas operações foi criada uma estrutura que inclui duas classes JAVA, `Doc.java` e `Grupo.java` e um EJB.

Descrição da Implementação

Desenho da Interface

Aqui apresenta-se a página inicial do sítio Web do estágio



Fig. A-17 Sítio Web Interno de Projecto - Interface

As próximas imagens ilustram a interface de operações de documentação

**Inserir Novo Documento:**

Ficheiro:

Descrição:

Grupo:

Fig. A-18 Sítio Web Interno de Projecto –  
Inserção de Documento



**Inserir Novo Grupo:**

Grupos existentes actualmente:

Apresentações

iPlanet Portal Server 3.0

Produtos iPlanet

BEA WebLogic Server 6.1

LDAP

Segurança

Programação

Apresentações

iPlanet Portal Server 3.0

Produtos iPlanet

BEA WebLogic Server 6.1

LDAP

Segurança

Programação

Apresentações

iPlanet Portal Server 3.0

Produtos iPlanet

BEA WebLogic Server 6.1

LDAP

Segurança

Programação

Nome do Novo Grupo:

OK

Fig. A-19 Sítio Web Interno de Projecto – Inserção de Novo Grupo

**Documentação disponível (working revisions):**

[ Apresentações ] [ iPlanet Portal Server 3.0 ] [ Produtos iPlanet ] [ BEA WebLogic Server 6.1 ]

[ LDAP ] [ Segurança ] [ Programação ]

**APRESENTAÇÕES**

Apresentação sobre o iPlanet Portal Server 3.0, BEA Weblogic 5.1 e Conceitos Associados

april\_20020322.pdf

**IPLANET PORTAL SERVER 3.0**

Manual de Administração do iPlanet Portal Server 3.0

admin\_guide\_20020306.pdf

Descreve a arquitectura global de sistema do IPlanet Portal Server 3.0

index\_iportal\_arch\_20020304.pdf

Notas de release do último service pack do iPlanet Portal Server

iportal\_sp4\_20020313.pdf

Guia de referência para programadores que dá exemplos de uso das API fornecidas

prog\_ref\_guide\_20020306.pdf

VirtualVault vs iPlanetPortal Server - Security Issues

vv\_iportal\_security\_issues.pdf

Fig. A-20 Sítio Web Interno de Projecto – Listagem da Documentação

65

A estrutura em termos de classes é apresentada a seguir:

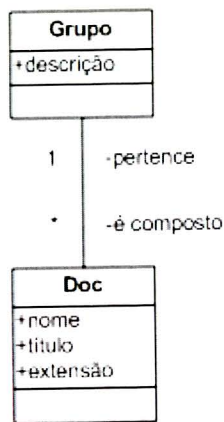


Fig. A-21 Sítio Web Interno de Projecto –  
Diagrama de Classes

Todos os métodos inerentes a estas páginas (inserções, listagens, criação de grupos) estão encapsulados num EJB, `DocsEJB`, que por sua vez utiliza um ficheiro de texto estruturado para guardar e ler as informações. O recurso a um ficheiro explica-se pela extrema simplicidade da estrutura de dados e pelo facto dos dados mudarem muito pouco ao longo do tempo.

Os métodos fornecidos pelo EJB são os seguintes:

```
public Vector getGrupos() throws RemoteException
```

Serve para listar todos os grupos existentes e respectivos ficheiros membros. No ficheiro de texto utilizador um grupo é delimitado pelo carácter “\*” no início e pelos caracteres “\*\*” no fim. Retorna um `Vector` de elementos do tipo `Group`.

```
public void criarGrupo(String nome) throws RemoteException
```

Cria um grupo, escrevendo para isso no ficheiro em modo sequencial.

```
public void escreverEmGrupo(String grupo, String nomeFich, String desc, String ext)
throws RemoteException
```

Escreve num dado grupo. Para este efeito é necessário abrir o ficheiro em modo aleatório e ir percorrendo até encontrar o grupo.



# ANEXO B – Planeamento do estágio e comentário

## Planeamento do Estágio

O planeamento inicial do estágio e encontra-se na tabela seguinte:

|    |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |                         |
|----|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------|
| 1. | Investigação das tecnologias a utilizar (Servidor Aplicaçonal WebLogic da BEA e iPlanet Portal Server da Sun)                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          | 4 Março<br>1 Abril      |
| 2. | Migração das tecnologias para uma máquina Solaris, cuja finalidade é a de servir o projecto nesta fase piloto. Esta fase consiste na instalação e configuração do seguinte software na máquina Solaris:<br><br>Oracle Client 8.1.7<br><br>BEA WebLogic Server 5.1 Service Pack 11<br><br>iPlanet Portal Server 3 Service Pack 4<br><br>Migração também para o servidor aplicaçonal da BEA toda a área WebParceiros, que é fundamentalmente a aplicação segura a que se pretende aceder.                                                                                                                                                                                                                                                                | 1 Abril<br><br>19 Abril |
| 3. | Trabalhar na solução single sign-on do portal, e implementar um método seguro de garantir que só determinadas máquinas físicas acedem realmente ao portal.<br><br>Nesta fase já se pode observar uma distinção mais precisa entre o trabalho do Joel Alvim e do Miguel Pereira.<br><br>O Joel Alvim ficou encarregado de planear os dados a serem replicados no servidor de perfis do portal, bem como implementar a solução Single Sign-On.<br><br>O Miguel Pereira irá concentrar-se mais na comunicação entre a base de dados Oracle, e o servidor LDAP. Paralelamente está também encarregue de desenvolver uma solução que garanta apenas o acesso ao portal, de determinadas máquinas físicas, o que irá mais além de verificar apenas o seu IP. | 19 Abril<br><br>2 Maio  |
| 4. | Migração do projecto para o sistema iPortal da Nexenter. Nesta fase o portal ainda irá utilizar ligações à base de dados Oracle.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       | 2 Maio<br><br>2 Junho   |
| 5. | Migrar todos os dados necessários da base de dados Oracle para o directório LDAP.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      | 2 Junho<br><br>Final    |

Como pode ver pela secção seguinte (Fases do Estágio), este planeamento foi seguido fielmente praticamente até ao ponto 3.

Na disponibilização da segunda máquina e acerto de configurações (encontram-se em sub-redes diferentes) também se gastou mais tempo que o esperado.

Finalmente, existiu a ramificação do projecto que levou mais de um mês e que não estava planeada.

## Fases e Momentos Importantes do Estágio

### Legenda:

Campos com sombreado – Acontecimentos importantes para o projecto.

Campos com texto a **negrito** – Fases do projecto.

| Descrição                                    |                                                        | Data de Início | Data de Fim | Comentários                                                                    |
|----------------------------------------------|--------------------------------------------------------|----------------|-------------|--------------------------------------------------------------------------------|
| Fase Inicial                                 |                                                        | 01-03-2002     | 01-04-2002  | Fase Inicial do Estágio                                                        |
|                                              | Documentação                                           |                |             |                                                                                |
|                                              | Formação                                               |                |             |                                                                                |
|                                              | Site de Projecto                                       |                |             |                                                                                |
| Disponibilização da Segnex                   |                                                        | 01-04-2002     |             |                                                                                |
| Primeira Reunião com Orientador da Faculdade |                                                        | 05-04-2002     |             |                                                                                |
| Implementação Portal de Serviços             |                                                        | 07-04-2002     | 04-06-2002  | Segunda Fase do Estágio                                                        |
|                                              | Viagem a Lisboa                                        | 07-04-2002     | 09-04-2002  |                                                                                |
|                                              | Instalação e Configuração                              |                |             | Preparação da Máquina Projecto Piloto                                          |
|                                              | Migração e Single Sign On WebParceiros                 | 19-04-2002     | 02-05-2002  | Passagem da aplicação para o Portal de Serviços                                |
|                                              | Investigação e Desenvolvimento da Solução de Segurança | 19-04-2002     | 10-05-2002  | Solução de identificação única da máquina cliente falada em reuniões da equipa |
|                                              | Integração 2ª Versão do Web Parceiros                  | 05-05-2002     |             |                                                                                |
|                                              | Mudança de Aspecto Gráfico                             | 11-05-2002     | 14-05-2002  | Aspecto Gráfico e Interface do Portal de Serviços                              |
|                                              | Página de Autenticação                                 |                |             |                                                                                |
|                                              | Desktop                                                |                |             |                                                                                |

|                                                                                  |                                                |                   |                   |                                                                                                        |
|----------------------------------------------------------------------------------|------------------------------------------------|-------------------|-------------------|--------------------------------------------------------------------------------------------------------|
|                                                                                  | Estrutura LDAP                                 | 15-05-2002        | 20-05-2002        | Estudo e Implementação da Estrutura LDAP mais adequada                                                 |
|                                                                                  | Integração da Applet                           | 20-05-2002        | 01-06-2002        | Integração da solução de identificação única da máquina cliente (versão JAVA) no Portal de Serviços    |
|                                                                                  | Módulo de Autenticação                         |                   |                   | Módulo de autenticação customizado usando a applet e a autenticação LDAP                               |
|                                                                                  | Activação                                      |                   |                   | Página de Activação de Utilizador, em que a applet e a JAVA Virtual Machine são instaladas             |
|                                                                                  | Feedback Reports                               | 23-05-2002        | 03-06-2002        | Desenvolvimento da aplicação de Feedback Reports, para os operadores telefónicos inserirem informações |
| Reunião da board de directores                                                   |                                                | 04-06-2002        |                   |                                                                                                        |
| <b>Aperfeiçoamento da solução Encontrada</b>                                     |                                                | <b>05-06-2002</b> | <b>15-06-2002</b> | <b>Fase Intermédia do Estágio</b>                                                                      |
|                                                                                  | Renovação da Documentação                      |                   |                   | Incluindo partes destinadas ao relatório de estágio                                                    |
|                                                                                  | Aperfeiçoamento da applet                      |                   |                   | Aperfeiçoamento da Versão JAVA                                                                         |
|                                                                                  | Aperfeiçoamento do Single Sign On WebParceiros |                   |                   |                                                                                                        |
| <b>Anúncio da Integração da solução de segurança no WebParceiros de PRODUÇÃO</b> |                                                | <b>16-06-2002</b> |                   |                                                                                                        |
| <b>Integração solução de segurança no WebParceiros</b>                           |                                                | <b>16-06-2002</b> | <b>01-08-2002</b> | <b>Ramificação ao projecto Portal de Serviços</b>                                                      |
|                                                                                  | Requisitos                                     |                   |                   |                                                                                                        |



|                   |                                                            |                   |                   |                                                                                           |
|-------------------|------------------------------------------------------------|-------------------|-------------------|-------------------------------------------------------------------------------------------|
|                   | Investigação da adaptação da solução de segurança          |                   |                   | Passagem de Tecnologia JAVA para ActiveX, por motivos de rapidez de instalação e execução |
|                   | Modificações WebParceiros                                  |                   |                   | Introdução das novas opções e novo tipo de <i>login</i>                                   |
|                   | Testes                                                     |                   |                   | Testes Unitários às novas opções                                                          |
|                   | Entrada de Jorge Sousa para projecto                       | 01-07-2002        |                   |                                                                                           |
|                   | Documentação da solução de segurança                       |                   |                   | Detail Design e General Design                                                            |
|                   | Entrada das novas opções para System Test                  | 22-07-2002        |                   |                                                                                           |
|                   | Resolução de problemas                                     |                   |                   | Problemas nas novas opções WebParceiros                                                   |
| <b>Fase Final</b> |                                                            | <b>01-08-2002</b> | <b>31-08-2002</b> | <b>Fase final do projecto</b>                                                             |
|                   | Visita a Lisboa                                            |                   |                   | Preparação da nova máquina que alojará o servidor aplicacional e o directory server       |
|                   | Disponibilização da nova máquina                           | 13-08-2002        |                   |                                                                                           |
|                   | Single Sign On SFAWeb                                      |                   |                   |                                                                                           |
|                   | Criação de novas classes gerais de apoio ao Single Sign On |                   |                   |                                                                                           |
|                   | Preparação para migração NEXENTER                          |                   |                   |                                                                                           |
|                   | Início da Migração                                         |                   |                   |                                                                                           |

# ANEXO C – Testes Unitários

Este anexo refere-se aos testes efectuados à integração descrita na secção 4.2. O procedimento padrão para a realização de testes na Vodafone Telecel S.A especifica que se pensem em funcionalidades (retiradas dos requisitos, normalmente) e que se tentem satisfazer todos os casos passíveis de acontecer numa utilização normal. Estes testes foram bastante extensos e extremamente importantes porque garantiram que a aplicação correspondia aos requisitos. Estes procedimentos incluíram testes em todos os sistemas operativos Windows (95,98,NT,2000,XP) de modo a garantir o funcionamento correcto nos vários tipos de *hardware* existente nos agentes.

A estrutura de cada teste é a seguinte:

|                                                        |
|--------------------------------------------------------|
| <b>Teste</b>                                           |
| <Descrição do teste>                                   |
| <b>Dados</b>                                           |
| <Dados relevantes do teste>                            |
| <b>Resultado:</b>                                      |
| <b>LOG</b>                                             |
| <Mecanismo de logging do servidor applicacional>       |
| <Tabelas da base de dados depois de realizado o teste> |

## Teste de Registo de Equipamento

### Teste

Registar um equipamento numa loja sem equipamentos, fornecendo um nome válido

### Dados

user\_id=69 (M\_VCALADO)  
sales\_point\_id=12 (C.Comercial Jumbo de Setúbal, Loja 10 2901 Setúbal Codex)  
nome do equipamento: PCTestel

### Resultado

#### LOG:

callBean.jsp: Created EJB PMSPEquipValidationEJB  
Função checkPC  
select ID FROM SP\_EQUIP\_VALIDATION e, SP\_SALES\_POINT s WHERE  
e.SALES\_POINT\_ID=s.SALES\_POINT\_ID AND e.SALES\_POINT\_ID='12' AND s.USER\_ID='69' AND  
e.EQUIP\_NAME='PCTestel'

```
Recebi um MAC :0000e258f90a

select ID from SP_EQUIP_VALIDATION where SERIAL_NUMBER IN ('0000e258f90a')

Função checkPC

select ID FROM SP_EQUIP_VALIDATION e, SP_SALES_POINT s WHERE
e.SALES_POINT_ID=s.SALES_POINT_ID AND e.SALES_POINT_ID='12' AND s.USER_ID='69' AND
e.EQUIP_NAME='PCTestel'

insert into SP_EQUIP_VALIDATION VALUES
('','12','0000e258f90a','PCTestel','69',SYSDATE,'69',SYSDATE)
```

PM\_USER\_ACTIVITY:

| OPTION_ID | CREATION_DATE       | DESCRIPTION                                    | PLACE                             |
|-----------|---------------------|------------------------------------------------|-----------------------------------|
| 180       | 22-07-2002 15:41:39 | Operação de registo foi concluída com sucesso; | registEquipamentoResult.service() |
| 180       | 22-07-2002 15:40:57 | Foi acedida a opção.                           | SAdapterSP.login()                |

SP\_EQUIP\_VALIDATION:

| SALES_POINT_ID | SERIAL_NUMBER | EQUIP_NAME | CREATION_USER | CREATION_DATE       | LAST_UPDATE_USER | LAST_UPDATE_DATE    |
|----------------|---------------|------------|---------------|---------------------|------------------|---------------------|
| 12             | 0000e258f90a  | PCTestel   | 69            | 22-07-2002 15:41:38 | 69               | 22-07-2002 15:41:38 |

Teste efectuado com SUCESSO

Teste

Nome de equipamento com caracteres inválidos

Dados

User\_id=69  
sales\_point\_id=12 (C.Comercial Jumbo de Setubal, Loja 10 2901 Setubal Codex)  
Nome do equipamento: Teste

Resultado

Aparece uma janela JavaScript que informa:  
Nome de equipamento inválido. Só são aceites [Aa-Zz]\_,.!@#\$\$%^&\*()+[0-9] e espaço

Teste efectuado com SUCESSO

Teste

Tentar registar equipamento já registado

Dados

user\_id=69



sales\_point\_id=10 (C.Comercial Fonte Deira, loja 60 - 7300 Portalegre)

Nome passado: PCTeste

Endereço MAC do equipamento: 00-02-A5-DE-7C-F8

Resultado

Verificação que o equipamento se encontra registado:

| Row # | ID  | SALES_POINT_ID | SERIAL_NUMBER | EQUIP_NAME | CREATION _USER | CREATION _DATE       | LAST_UPDATE_ USER | LAST_UPDATE_ DATE    |
|-------|-----|----------------|---------------|------------|----------------|----------------------|-------------------|----------------------|
| 7     | 288 | 13             | 0002a5de7cf8  | Miguel     | 69             | 22-Jul-2002 14:41:29 | 69                | 22-Jul-2002 14:41:29 |

Registo do equipamento na loja sales\_point\_id=10 (C.Comercial Fonte Deira, loja 60 - 7300 Portalegre):

1º passo

LOG:

Função checkPC

```
select ID FROM SP_EQUIP_VALIDATION e, SP_SALES_POINT s WHERE
e.SALES_POINT_ID=s.SALES_POINT_ID AND e.SALES_POINT_ID='10' AND s.USER_ID='69' AND
e.EQUIP_NAME='PCTeste'
```

Query retornou 0 resultados!

Fim da Função checkPC

2º passo

Carregamento da página com o controlo ActiveX, passando depois para a detecção automática e mensagem de erro sem intervenção do utilizador

3º passo

LOG:

Recebi um MAC :0002a5de7cf8

```
select ID from SP_EQUIP_VALIDATION where SERIAL_NUMBER IN ('0002a5de7cf8')
```

Já existe um equipamento com pelo menos um dos MACS passados

```
registEquipamentoResult.jsp PMSPEException: PMSPEException : Error code = 2000
Place=activate()
```

```
PMSPUtils.writeLog with no highArgs (2000, registEquipamentoResult.jsp optionId =
180
```

4º Passo

Detecção automática de equipamento já registado (e informação sobre o equipamento já existente) e mensagem de erro sem intervenção do utilizador.

Este Equipamento já se encontra registado com o nome ola2 na loja C.Comercial Jumbo Alfragide, lj.10 - 2700 Amadora. (mensagem 2000)

PM\_USER\_ACTIVITY:

| Row # | OPTION _ID | CREATION_ DATE       | DESCRIPTION                                                           | PLACE                       |
|-------|------------|----------------------|-----------------------------------------------------------------------|-----------------------------|
| 1     | 180        | 22-Jul-2002 17:36:17 | O Equipamento do Utilizador M_VCALADO já se encontra na base de dados | registEquipamentoResult.jsp |

Teste efectuado com SUCESSO

Teste

Registar correctamente em plataforma XP, NT e Windows95/98

Dados

User\_id=69

Sales\_point\_id=13 (C.Comercial Jumbo Alfragide, lj.10 - 2700 Amadora)

Equipamentos registados:

- Windows NT4 Service Pack 5 + Internet Explorer 5.0 + 1 placa de rede
  - o teste\_nt
- Windows NT4 Service Pack 5 + Internet Explorer 5.5 + 1 placa de rede
  - o aaaa
- Windows XP + Internet Explorer 6.0 + 2 placas de rede
  - o jsilva
- Windows NT4 Service Pack 3 + Internet Explorer 5.0 + 1 placa de rede
- Windows 98 + Internet Explorer 5.0 + 1 placa de rede
- Windows 95 + Internet Explorer 5.0 + 1 placa de rede

Resultado

- teste\_nt (Windows NT4 Service Pack 5 e Internet Explorer 5.0 + 1 placa de rede)

PM\_USER\_ACTIVITY:

| Row # | ID    | OPTION_ ID | ACTIVITY_ ID | USER_ ID | TARGET_ USER_ID | CREATION _DATE       | DESCRIPTION                                   | PLACE                             |
|-------|-------|------------|--------------|----------|-----------------|----------------------|-----------------------------------------------|-----------------------------------|
| 5     | 22276 | 200        | 2005         | 69       | 69              | 24-Jul-2002 12:48:32 | Operação de registo foi concluída com sucesso | registEquipamentoResult.service() |

SP\_EQUIP\_VALIDATION:

| Row # | ID | SALES_POINT_ID | SERIAL_NUMBER | EQUIP_NAME | CREATION_USER | CREATION_DATE           | LAST_UPDATE_USER | LAST_UPDATE_DATE        |
|-------|----|----------------|---------------|------------|---------------|-------------------------|------------------|-------------------------|
| 920   | 13 |                | 0001030ebb9c  | teste_nt   | 69            | 24-Jul-2002<br>12:48:32 | 69               | 24-Jul-2002<br>12:48:32 |

– jsilva (Windows XP e Internet Explorer 6.0 + 2 placas de rede)

PM\_USER\_ACTIVITY:

| Row #  | ID | OPTION_ID | ACTIVITY_ID | USER_ID | TARGET_USER_ID | CREATION_DATE           | DESCRIPTION                                   | PLACE                             |
|--------|----|-----------|-------------|---------|----------------|-------------------------|-----------------------------------------------|-----------------------------------|
| 322279 |    | 200       | 2005        | 69      | 69             | 24-Jul-2002<br>12:49:40 | Operação de registo foi concluída com sucesso | registEquipamentoResult.service() |

SP\_EQUIP\_VALIDATION:

| Row # | ID | SALES_POINT_ID | SERIAL_NUMBER | EQUIP_NAME | CREATION_USER | CREATION_DATE           | LAST_UPDATE_USER | LAST_UPDATE_DATE        |
|-------|----|----------------|---------------|------------|---------------|-------------------------|------------------|-------------------------|
| 1121  | 13 |                | 00d059646ac9  | jsilva     | 69            | 24-Jul-2002<br>12:49:40 | 69               | 24-Jul-2002<br>12:49:40 |
| 1222  | 13 |                | 0002b32cbf27  | jsilva     | 69            | 24-Jul-2002<br>12:49:40 | 69               | 24-Jul-2002<br>12:49:40 |

– aaaa (Windows NT4 Service Pack 5 e Internet Explorer 5.5 + 1 placa de rede)

PM\_USER\_ACTIVITY:

| Row #   | ID | OPTION_ID | ACTIVITY_ID | USER_ID | TARGET_USER_ID | CREATION_DATE           | DESCRIPTION                                   | PLACE                             |
|---------|----|-----------|-------------|---------|----------------|-------------------------|-----------------------------------------------|-----------------------------------|
| 2522250 |    | 200       | 2005        | 69      | 69             | 24-Jul-2002<br>12:40:12 | Operação de registo foi concluída com sucesso | registEquipamentoResult.service() |

SP\_EQUIP\_VALIDATION:

| Row # | ID | SALES_POINT_ID | SERIAL_NUMBER | EQUIP_NAME | CREATION_USER | CREATION_DATE           | LAST_UPDATE_USER | LAST_UPDATE_DATE        |
|-------|----|----------------|---------------|------------|---------------|-------------------------|------------------|-------------------------|
| 518   | 13 |                | 00508b516666  | aaaa       | 69            | 24-Jul-2002<br>12:40:12 | 69               | 24-Jul-2002<br>12:40:12 |

– Windows NT4 Service Pack 3 + Internet Explorer 5.0 + 1 placa de rede

Tudo correu sem problemas. Foi usado o nome aaaa mas como este já se encontrava registado apareceu a mensagem “Este equipamento já se encontra registado”

- Windows 98 + Internet Explorer 5.0 + 1 placa de rede

Idem.

- Windows 95 + Internet Explorer 5.0 + 1 placa de rede

Foram feitas modificações por causa do Win95 porque este não possui por omissão as bibliotecas necessárias para a leitura do MAC.

Assim, é lido sempre o Volume Serial Number.

**SP\_EQUIP\_VALIDATION:**

| Row # | ID | SALES_POINT_ID | SERIAL_NUMBER | EQUIPMENT | CREATION_USER | CREATION_DATE        | LAST_UPDATE_USER | LAST_UPDATE_DATE     |
|-------|----|----------------|---------------|-----------|---------------|----------------------|------------------|----------------------|
| 151   | 13 |                | 487985885     | w95       | 69            | 28-Jul-2002 17:26:36 | 69               | 28-Jul-2002 17:26:36 |

**Teste efectuado com SUCESSO**







FACULDADE DE ENGENHARIA

UNIVERSIDADE DO PORTO

BIBLIOTECA



000066318