



# Do more with less - A framework for e-government

André Amaral Dâmaso Rodrigues Alves

## Orientadores

Doutora Arminda Guerra Lopes

Dissertação apresentada ao Instituto Politécnico de Castelo Branco para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Desenvolvimento de Software e Sistemas Interactivos, realizada sob a orientação científica do Doutor Arminda Guerra Lopes Professor Doutor do Departamento de Informática do Instituto Politécnico de Castelo Branco



## Composição do júri

Presidente do júri

Vogais

## **Agradecimentos**

Agradeço especialmente às pessoas que tornaram possível a concretização deste trabalho.

Á minha família por me apoiar e motivar na decisão de iniciar esta etapa e pelo interesse manifestado durante o desenvolvimento.



## Resumo

A missão dos serviços de apoio aos idosos é permitir que os nossos idosos possam viver com dignidade e proporcionar o apoio necessário para que consigam promover o seu sentimento de pertença, segurança e de reconhecimento. De igual modo, a segurança social Portuguesa, a entidade responsável por supervisionar as atividades das instituições sem fins lucrativos não-públicas para a solidariedade social tem de melhorar a qualidade das operações e serviços, e o processo de interação dos requisitos e informações do sistema alinhado com as dos centros sociais para pessoas idosas.

O principal objetivo deste trabalho é cumprir o que o próprio conteúdo do título diz, fazer muito com pouco, ou seja, significa projetar uma estrutura que vai agregar os sistemas de informação de diferentes centros sociais para os idosos. De acordo com algumas observações e entrevistas constatou-se que, hoje em dia, as instituições de solidariedade social não interagem, em termos de usabilidade, com os centros sociais para idosos. Quase todas as instituições utilizam diferentes sistemas de tecnologia da informação para a gestão, contabilidade e para outras funcionalidades. Por outro lado, os funcionários dessas instituições têm diferentes origens e experiências e, normalmente, eles não estão motivados com a tecnologia. Em situações como estas em que é frequente, eles necessitem de comunicar através de um número diversificado de formulários e documentos trocados.

O protótipo proposto será uma ferramenta que irá servir o processo de comunicação entre os centros sociais para idosos e a instituição de segurança social em termos de usabilidade.

## Palavras-chave

Centros sociais para idosos; instituto de segurança social, sistemas de informação, tecnologia da informação, usabilidade, e-government.



## **Abstract**

The mission of elderly services is to enable our elders to live in dignity and to provide necessary support for them to promote their sense of belonging, sense of security and sense of worthiness. Equally, the Portuguese social security, the entity which goals are to oversee the activities of non-profit non-public institutions for social solidarity has to improve the quality of operations and services, and the interaction process of the collection, requirements and information system aligned with those of the social centers for elderly.

The main goal of this paper is to fulfill the title content which means to design a framework that will aggregate the information systems of different social centers for the elderly. Accordingly to some observations and interviews it was found that, nowadays, the social solidarity institutions do not interact, in a usability terms, with the social centers for elderly. Almost all the institutions use different information technologies systems for management, accounting, and for other concerns. Conversely, the employees of these institutions have different backgrounds and experiences and normally they are not motivated with technology. In situations where, and this is frequently, they need to communicate, a diverse number of forms and documents are exchanged.

The proposed framework will be a tool to serve the communication process among the social centers for elderly and the national social security institution in terms of usability.

## **Keywords**

Social centers for elderly; social security institution, information systems, information technology, usability, e-government.



# Índice geral

## Índice

<b>1</b>	<b>Introdução</b> .....	<b>1</b>
1.1	Âmbito.....	1
1.2	Objetivos.....	2
1.3	Plano de trabalhos e cronograma.....	3
<b>2</b>	<b>Enquadramento teórico e contextual</b> .....	<b>5</b>
2.1	Gestão de IPSS (Instituição Particular de Solidariedade Social) /Misericórdias .....	5
2.2	O sistema de segurança social .....	7
2.3	Estruturas residenciais para idosos.....	9
2.4	Registos de utentes .....	11
2.5	E-government .....	14
<b>3</b>	<b>Metodologia</b> .....	<b>17</b>
<b>4</b>	<b>Caso de Estudo</b> .....	<b>21</b>
4.1	Centro Social Amigos da Lardosa (CSAL) .....	22
4.2	Análise de Dados.....	26
<b>5</b>	<b>Desenvolvimento tecnológico</b> .....	<b>30</b>
5.2.1	Internet information Services (IIS) .....	34
5.1.2	SQL Server .....	36
5.1.3	Crystal Reports.....	38
5.1.4	ASP.NET .....	39
5.1.5	Hyper Text Markup Language (HTML).....	41
5.1.6	Lightswitch .....	44
5.1.7	A Linguagem de Programação C#.....	46
5.1.8	Visual Basic (VB).....	48
<b>6</b>	<b>Arquitetura do Sistema</b> .....	<b>52</b>
<b>6.1</b>	<b>Modelos de Entidades e relacionamentos</b> .....	<b>53</b>
6.1.2	Entidades .....	54
<b>7</b>	<b>Funcionalidades implementadas</b> .....	<b>66</b>
<b>8</b>	<b>Testes de Validação</b> .....	<b>89</b>
<b>9</b>	<b>Conclusões</b> .....	<b>90</b>
<b>10</b>	<b>Trabalho Futuro</b> .....	<b>92</b>
<b>11</b>	<b>Referências</b> .....	<b>93</b>

## Índice de figuras

Figura 1 - Imagem que ilustra os dados em gráfico da Carta Social.....	9
Figura 2 - Esquema com as diferentes entidades do e-government.....	15
Figura 3 - Esquema das diferentes etapas da metodologia Action Research.....	19
Figura 4 - Estrutura hierárquica do Centro Social Amigos da Lardosa .....	22
Figura 5 - Esquema de funcionamento do Software existente.....	28
Figura 6 - Esquema de Software F3M .....	32
Figura 7 - Esquema ilustrativo de funcionamento do IIS. ....	35
Figura 8 - Esquema ilustrativo do funcionamento do ASP.net.....	40
Figura 9 - Esquema de funcionamento do Lightswitch.....	45
Figura 10 - Arquitetura do Software proposto na Tese.....	52
Figura 11 - Caso de uso Administrador.....	56
Figura 12 - Caso de uso Médico.....	56
Figura 13 - Caso de uso Enfermeiro .....	57
Figura 14 - Caso de uso Psicólogo .....	57
Figura 15 - Caso de uso Sociólogo .....	58
Figura 16 - Caso de uso Recepção.....	58
Figura 17 - Caso de uso Farmácia.....	59
Figura 18 - Caso de uso Apoio Social.....	59
Figura 19 - Diagrama de Sequência Autenticação.....	61
Figura 20 - Diagrama de Sequência criar utente .....	62
Figura 21 - Diagrama de Sequência Registrar Alteração.....	63
Figura 22 - Diagrama de Sequência criar instituição.....	64
Figura 23 - Diagrama de Sequência registo de monitorização.....	65
Figura 24 - Ecrã de login .....	66
Figura 25 - Procurar Cliente .....	67
Figura 26 - Lista de espera .....	67
Figura 27 - Criar Cliente .....	68
Figura 28 - Características físicas e funcionais .....	68
Figura 29 - Capacidade locomotora.....	69
Figura 30 - Capacidade mental.....	69
Figura 31 - Dependência física.....	70
Figura 32 - Admissões.....	70
Figura 33 - Criar admissões.....	71
Figura 34 - Dados de monitorização.....	71
Figura 35 - Terapêutica de enfermagem .....	72
Figura 36 - Registo de Enfermagem .....	72
Figura 37 - Registo de psicologia.....	73
Figura 38 - Tabela de refeições .....	73
Figura 39 - Tabela de consistência.....	74
Figura 40 - Lista de refeições .....	74
Figura 41 - Registos de apoio social .....	75
Figura 42 - lista de bens do utente.....	75
Figura 43 - gestão do cofre.....	76
Figura 44 - entrada e saída do cofre .....	76
Figura 45 - criar valências .....	77
Figura 46 - criar instituições .....	77
Figura 47 - criar tipo de instituições .....	78

Figura 48 - participações financeiras.....	78
Figura 49 - criar participação.....	79
Figura 50 - criar funcionário.....	79
Figura 51 - clientes por instituições.....	80
Figura 52 - lista de funcionários.....	80
Figura 53 - funcionários por categoria.....	81
Figura 54 - documentos dos funcionários.....	81
Figura 55 - registos biométricos.....	82
Figura 56 - definir horários.....	82
Figura 57 - criar fornecedor.....	83
Figura 58 - criar fatura.....	83
Figura 59 - criar produto.....	84
Figura 60 - criar recibo.....	84
Figura 61 - criar taxa de iva.....	85
Figura 62 - Criar compras.....	85
Figura 63 - lista de recursos por utente.....	86

## Lista de Tabelas

Tabela 1 - Controle de clientes vs controle de utilizadores .....	24
Tabela 2 - Contabilidade gerencial.....	25
Tabela 3 - Arquitetura de Zachamn .....	26

## **Lista de abreviaturas, siglas e acrónimos**

API – Applications programming interface  
AR – Action Research  
ASMX – Active Server Methods  
ASP – Active Server Pages  
ASPX – Active Server Pages framework  
CATL – centro de actividades de tempos livres  
CIL – Common intermediate Language  
CLI – Common Language interface  
CLR – Common Language Runtime  
CoM – component object model  
COTEC - Associação Empresarial para a Inovação  
CSAL – centro social amigos da Lardosa  
CSS – Cascading Style sheet  
DLL – dynamic link library  
DMV – dynamic management views  
ERPI – Estrutura residencial para idosos.  
FTP – File transfer protocol  
GUI – Graphical user interface  
HTML – Hypertext markup Language  
IDE – integrated drive electronics  
IIS – Internet Information Services  
IPSS – Instituição Particular de Solidariedade Social  
ISO – International standard organization  
JSP – java server pages  
Ldf – local development framework  
MVC – model view controller  
PHP – Hypertext Preprocessor  
PME – Pequena média empresa  
PWS – Personal Web Server  
RDBMS – relational Database management system  
SAD – serviço de apoio domiciliário

SCML – Santa casa da Misericórdia de Lisboa

SOAP – simple object access protocol

SQL – structured query Language

TDS – tabular data Stream

TIC – Tecnologias de informação e comunicação

UDT – user defined type

VBA – visual basic for Applications

WCF – Windows communication Foundation



# 1 Introdução

## 1.1 Âmbito

As Instituições Particulares de Solidariedade Social (IPSS), organizações da economia social, especializadas na ação social, refletem um papel social muito importante para o Estado e para a sociedade civil. São instituições sem fins lucrativos geridas por particulares. No entanto, o estado e municípios contribuem com algum orçamento o qual permite manter algumas das instituições a trabalhar, pois a maioria dos clientes têm rendimentos baixos. A gestão destas instituições é difícil, por um lado, porque os fundos são provenientes de fontes diferentes (doações, subsídios da segurança social, do estado e outros) e, porque do ponto de vista de processamento de informações, há várias fontes em diferentes tipos de suporte (relatórios, documentos, os resultados das aplicações), o que torna impossível a atualização e pesquisa.

As instituições de solidariedade social enfrentam diversas adversidades o que nos motivou a investigar esta área especialmente no que diz respeito aos sistemas de informação e aplicações da tecnologia. Estes são usados acima de suas capacidades e os utilizadores têm pouca experiência nestas áreas.

O objetivo deste trabalho é apresentar um caso de estudo, onde através de observações e entrevistas, a informação foi recolhida e analisada através de métodos qualitativos permitindo-nos construir um quadro, como uma imagem de boas práticas, a ser usado para comunicação entre as instituições de solidariedade social e os diversos lares de idosos.

Depois de ser efetuado o quadro com as boas práticas referidas anteriormente, é efetuado o desenvolvimento de uma aplicação que pode ser usada, como exemplo, para conciliar todos os objetivos que uma aplicação para o sector envolvido deve conter.

## 1.2 Objetivos

O objetivo da presente dissertação é o desenvolvimento de uma plataforma aplicacional baseada em tecnologias Microsoft, com uma arquitetura modular e extensível, que substitua um leque de ferramentas e utilitários dispersos atualmente utilizados nas IPSS e Segurança Social. Tal aplicação terá como objetivo permitir a otimização do funcionamento da organização e auxiliar no processo de gestão da mesma dando um foco maior na gestão do utente, uma vez que se trata do módulo com mais importância e que mais especificidade apresenta:

- **Gestão de utentes:** Deverá dar suporte á gestão de utentes, tanto inscritos como já pertencentes á instituição, de forma a controlar as listagens de espera e de tratamento de dados dispondo de informação e evolução dos mesmos.
- **Gestão de Recursos Humanos:** Poder controlar todos os recursos humanos existentes nas instituições, assim como documentação, horários e vencimentos.
- **Gestão da instituição:** Deve ser possível gerir toda a instituição de forma a poder ser controlado todos os dados gerais da mesma, tais como vagas por resposta social, vagas totais, rendimentos mensais e anuais assim como complementos vindos da Segurança Social.
- **Gestão de armazém:** Deverá ser possível controlar todo o stock da instituição de forma a poder saber as entradas e saídas e verificar alertas para material necessário assim como os custos associados a cada um.
- **Gestão da Segurança Social:** Através do utilizador da segurança social é possível auditar as instituições de forma mais rápida e eficiente uma vez que a plataforma trabalha sobre a forma web consegue-se sem sair do edifício controlar as diversas instituições podendo auxiliar na tomada de decisões.

### 1.3 Plano de trabalhos e cronograma

Para atingir os objetivos acima identificados, estipulou-se o seguinte plano de trabalhos, cujas etapas se descrevem de seguida:

**Etapa 1** – Estudo do trabalho relacionado e delimitação do problema: Esta etapa destina-se ao trabalho relacionado, com particular foco nas principais aplicações concorrentes, que dão resposta às necessidades específicas da organização.

**Etapa 2** – Definição de requisitos: Esta etapa foca-se na análise do objetivo da aplicação, concretamente nas suas características, nas restrições de utilização e nos seus utilizadores finais.

**Etapa 3** – Implementação, teste e validação: Após a fase inicial de estudo do trabalho relacionado e de levantamento de requisitos funcionais, terá início a fase de desenvolvimento. Procurar-se-á promover um constante contacto incluindo testes e validação com o utilizador final sempre que seja possível e necessário.

**Etapa 4** – Implantação: Etapa destinada ao processo de integração da aplicação em ambiente de produção.

**Etapa 5** – Escrita e revisão da dissertação: A redação da dissertação decorre das etapas anteriores, e inclui uma fase de revisão do documento final.



## 2 Enquadramento teórico e contextual

### 2.1 Gestão de IPSS (Instituição Particular de Solidariedade Social) /Misericórdias

O sistema de cuidados de longa duração, em Portugal, não participou do envolvimento do setor público, até há pouco tempo, as Misericórdias são tipos de instituições independentes e sem fins lucrativos, têm como base uma decisão do foro religioso.

Nas últimas décadas, a importância de apoiar os idosos dependentes foi destacada e reforçada através da intensificação e melhoria dos serviços de saúde. Em 2006, devido ao crescente número de idosos e a escassez de serviços, a Rede Nacional de Cuidados Continuados Integrados foi implementada, com base na estrutura existente. Em cada região portuguesa, os serviços sociais são fornecidos pelo Ministério do Trabalho e da Solidariedade Social. A saúde e assistência social são fornecidos principalmente por organizações privadas sem fins lucrativos (instituições subsidiadas pelo Estado) e Misericórdias.

Hoje em dia, os serviços e valências disponíveis para os idosos são várias: centros de dia, centros de cuidados domiciliários, serviços de apoio domiciliário e cuidados domiciliários integrados que incluem cuidados de saúde, enfermagem, limpeza das casas e cuidados paliativos para pessoas altamente dependentes, também cuidados residenciais e acomodação familiar. No entanto, as duas últimas soluções são ainda muito pouco desenvolvidos.

A fim de expandir os serviços de um novo mix público / privado centrado no público de subsidiar instituições sem fins lucrativos foi construído no final de 1980. Foi implementado através de uma nova legislação sobre o estatuto jurídico das instituições sem fins lucrativos (designado como IPSS - Instituições Particulares de Solidariedade Social) e a institucionalização das negociações anuais e acordos, entre os representantes do terceiro setor e do ministério encarregado dos assuntos sociais, sobre o subsídio fixo pago pelo Estado por pessoa idosa cuidada por serviços.

As instituições chamadas Instituições Particulares de Solidariedade Social (IPSS) emergem a partir da iniciativa de particulares ou associações, que são não-lucrativas e têm como objetivo a solidariedade social, este tipo de instituições são reconhecidas pelo Estado e subsidiadas pelo mesmo.

Em termos de sistemas de informação estas instituições enfrentam vários problemas. As novas tecnologias não são usadas adequadamente para a gestão, e as empresas de desenvolvimento de software têm aplicações individuais para cada tarefa a ser executada: gestão de utilizadores, contabilidade, gestão de veículos e assim por diante. Plataformas que cobrem todos eles não existem. Consequentemente, várias horas estão a ser gastas na má usabilidade das aplicações que poderiam ser utilizadas

para outras funções. Por outro lado, a instituição de segurança social exige vários relatórios para as instituições que são difíceis de preparar já que a informação está espalhada por escritórios.

Para as IPSS, os sistemas de gestão da qualidade são apresentadas pelas instituições de solidariedade social como uma diretriz de funcionamento importante, com todas as regras que devem ser cumpridas. No entanto, existem burocracias a seguir a cada vez que uma intervenção é feita com o cliente e que leva demasiado tempo tendo em conta a quantidade de informação a processar.

## 2.2 O sistema de segurança social

O sistema de segurança social em Portugal é administrado pelo Estado e, em princípio, aplicável a todos os indivíduos que trabalham em Portugal, quer como empregados por conta de outrem ou autónomos. Este sistema oferece benefícios para a saúde, doença, reforma, invalidez, morte, velhice, maternidade, paternidade e adoção.

A relação entre o sistema de segurança social e as instituições particulares de solidariedade social (IPSS) é extremamente importante para o seu funcionamento. A segurança social faz, anualmente, os acordos entre as instituições, os protocolos de cooperação, a fim de atender as necessidades da instituição. Além destes protocolos, a segurança social elabora manuais de qualidade com as orientações para ajudar as instituições a criar o seu próprio manual da qualidade.

Os manuais de qualidade são organizados em valências: abrigo Familiar, Casa Residencial, Centro de Atividades Ocupacionais, Casa da Infância e Juventude, Centro Residencial de Acolhimento Temporário, Estruturas para Idoso, Creches, e serviços de apoio domiciliário.

O objetivo desses modelos são:

- Um instrumento para a autoavaliação das Respostas Sociais, permitindo a revisão sistemática do seu desempenho;
- Apoiar o desenvolvimento e a implementação de um sistema de gestão da qualidade, permitindo uma melhoria significativa da sua organização e funcionamento;
- Adicionar, num referencial normativo, todos os requisitos aplicáveis a uma Resposta Social, independentemente da natureza jurídica da instituição.

Este modelo foi apoiado por alguns instrumentos manuais: o processo-chave e questionários de avaliação de clientes, colaboradores e acionistas.

O compromisso da segurança social e o de todos os responsáveis nos diversos níveis, é promover respostas qualificadas para os cidadãos, de forma partilhada e responsável. É descrito no manual de qualidade da estrutura residencial para idosos que "Assim, é hora de ajustar as respostas sociais à nova realidade, contribuindo com as políticas públicas para um exercício de cidadania mais responsável, assumindo os seguintes princípios políticos:

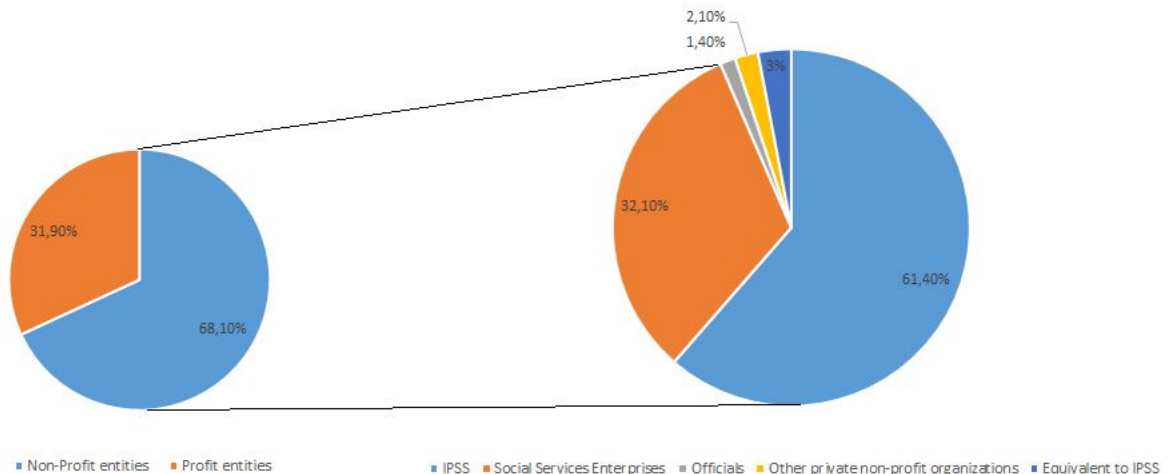
- Diferenciar as respostas, de acordo com as condições particulares dos seus destinatários, com as circunstâncias próprias dos distintos territórios, respeitando a equidade na distribuição dos recursos.
- Estipular as soluções assegurando que todos os intervenientes, cidadãos, famílias, públicas e institutos privados estão mobilizados e que eles assumem compromissos nas intervenções em que eles participam.

Com a implementação de manuais de qualidade o papel do auditor de segurança social é facilitado, uma vez que atribui classificações de acordo com os parâmetros que serão alcançados pelas instituições. Para as instituições, o seu controlo e a tarefa de apresentar resultados e registos será facilitada se um quadro for desenhado para isso. Hoje em dia, a informação está espalhada e os recursos humanos estão com medo da quantidade de atividades e as exigências da segurança social.

Um framework de desenvolvimento poderá responder às necessidades dos problemas encontrados, em primeiro lugar, deverá reunir todas as áreas e as informações necessárias para o centro com as necessidades integrais que podem ser realizadas na instituição.

## 2.3 Estruturas residenciais para idosos

Há dois grupos principais de lares em Portugal: as entidades sem fins lucrativos e as instituições com fins lucrativos. A posterior inclui as instituições de solidariedade social, chamadas de instituições particulares de solidariedade social (IPSS), os serviços sociais de empresas e da "Misericórdia" (SCML). A sua distribuição, em Portugal é apresentada na figura 1.



**Figura 1** - Imagem que ilustra os dados em gráfico da Carta Social

Em 31 de Dezembro de 2010, e de acordo com a 'Carta Social', havia cerca de 6.000 instituições de solidariedade social. Cerca de 70 por cento eram instituições sem fins lucrativos e 62 por cento eram instituições particulares de solidariedade social.

As atividades que decorrem de movimentos sociais e económicas de natureza associativa são integradas no sector social, económicas. Esse novo sector está situado entre o sector privado com fins lucrativos e o sector público.

Vaz (2007) considera que o sector terciário vem de uma tradição anglo-saxónica, onde organizações sem fins lucrativos estão incluídas. Na forma jurídica do sector sem fins lucrativos, também conhecido como sector voluntário, está particularmente ligada ao contexto americano Nord, onde a relação é feita dentro de uma tradição do estado social.

Em Portugal, a economia social é dividida em três subsectores:

- Cooperativo - economia mercantil, que compreende os "sectores de atividade onde a intensidade capitalista era fraco" (Laville, 2000:532 citado por França Filho, 2002);
- Associativo - jovem, divertido, associações culturais, desportivas e ONG (organizações não-governamentais);

- Social - representado por IPSS.

A economia social está em desenvolvimento, nacional e internacional. Ela cresce e floresce com as crises do Estado e com a intensificação das necessidades sociais, são uma grande contribuição para a diminuição da taxa de desemprego.

As principais áreas onde a economia social trabalha são os serviços sociais, os serviços que contribuem para melhorar a qualidade de vida, os serviços culturais e de entretenimento e os serviços ambientais.

De acordo com a informação social do instituto da segurança, o número de trabalhadores e utentes, em Portugal, tem aumentado consideravelmente, especialmente nos últimos vinte anos. Este número vai crescer devido ao envelhecimento da população e da exclusão social.

O governo reconhece a relevância de IPSS na prestação de serviços sociais à população através do estabelecimento de acordos de cooperação e financeiras. Na verdade, o apoio familiar tem vindo a diminuir e o Estado considera as IPSS uma parte estratégica no sistema de cuidados.

Comparando com o cuidado residencial fornecido pelo setor público, os lares dirigidos por Misericórdias e outras instituições sem fins lucrativos são geralmente de melhor qualidade e apenas solicitam uma contribuição nominal de pacientes e das suas famílias. Os lares de idosos no setor privado são muito caros e a maioria da população não pode pagar por eles.

O apoio domiciliário está a expandir-se em Portugal. Em algumas infraestruturas das regiões, para oferecer apoio ao idoso, foram desenvolvidas em parceria com os municípios, as administrações regionais de saúde e instituições sem fins lucrativos. Aparentemente, o estabelecimento de redes de assistência social está a tornar-se uma prioridade.

## 2.4 Registos de utentes

Nesta secção é descrita a metodologia de registo de utentes, para que os procedimentos venham a ser considerados no desenvolvimento da aplicação.

Um registo consiste na documentação sistematizada do historial clínico e de cuidados de saúde prestados a um determinado paciente. É um conjunto ordenado de documentos que contêm dados clínicos e pessoais de um paciente, registados pelos profissionais de saúde.

O registo clínico é usado na prática clínica como a base do planeamento da prestação de cuidados de saúde de um dado paciente, e servindo como plataforma de comunicação e partilha de informação entre médicos e outros profissionais do ramo (quando o registo não pertence a um só médico). Adicionalmente, os registos médicos também podem servir para fins científicos (pesquisa e desenvolvimento da medicina), pedagógicos (estudantes de medicina/médicos residentes), e de gestão administrativa (p.ex. fornecer dados estatísticos para análise de procedimentos clínicos e auditorias a serviços).

Estes registos contêm considerações, achados, resultados de meios complementares de diagnóstico e informações sobre o tratamento do processo patológico. Os grupos de dados podem ser de:

- Carácter administrativo (nome, data de nascimento, Bilhete de Identidade, etc.);
- Dados médicos fixos de ocorrência singular (p.ex. grupo sanguíneo, dados biométricos, etc.);
- Dados médicos variáveis de ocorrência múltipla (p.ex. história clínica, diagnósticos, procedimentos, terapêutica, etc.).

Complementarmente, um registo pode ainda conter informação proveniente de outras fontes, mas que habitualmente apenas é referenciada (tem que ser explicitamente solicitada), não estando disponível juntamente com o ficheiro físico. Isto acontece com registos médicos em papel, sendo que já com os registos eletrónicos a informação clínica de um dado paciente está disponível como um todo.

Em Portugal a ficha clínica assim como a decisão da informação que dela consta, pertence ao médico que a criou, sendo que por lei, o clínico é obrigado a revelar a informação ao respetivo paciente (embora não seja obrigado a partilhá-la ou transferi-la para outros clínicos, a não ser que se trate do serviço público de saúde. Nesse caso existe mesmo um modelo de ficha clínica). Na medicina privada, o médico poderá ter de que fazer uma carta com o resumo do historial clínico do paciente para entregar a outro médico, mas a ficha do paciente pertence-lhe.

De acordo com a administração central dos sistemas de Saúde, existem tipicamente três tipos de organização do registo clínico:

- *Time-oriented medical record* – os dados e observações são registadas de forma cronológica. Este tipo de organização é característica da medicina efetuada antes de se realizarem meios complementares de diagnóstico, em que o que o médico regista aquilo que o doente lhe conta e aquilo que ele observa;

- *Source-oriented medical record* – os dados clínicos são organizados consoante a sua proveniência. A ordem é dada pela origem da informação (p.ex. história clínica, resultados dos meios complementares de diagnóstico);

- *Problem-oriented medical record* – os dados clínicos são organizados por problema. Este tipo de organização da informação clínica surgiu nos anos 60, sendo que para cada problema os dados são organizados de acordo com a estrutura SOAP:

- S – Subjetivo – Dados da história clínica da narrativa do paciente;
- O – Objetivo – Dados do exame físico e resultados de exames auxiliares de diagnóstico;
- A – Avaliação – Conclusões e diagnóstico;
- P – Plano – Plano médico, tratamento ou atitude.

As vantagens dos registos eletrónicos sobre as tradicionais fichas em papel, em termos de gestão e flexibilidade e rapidez de acesso à informação, são evidentes. No entanto destacam-se os seguintes pontos:

- Acesso simultâneo à informação do paciente – a ficha de um paciente pode ser consultada e editada por vários profissionais simultaneamente, mesmo em locais físicos diferentes (desde que o sistema assim o permita, uma vez que por razões de segurança ou de pertença do dados clínicos, o sistema poderá não ser acedido fora das redes locais);

- Legibilidade do registo – os tradicionais registos em papel são manuscritos, sendo muitas vezes ilegíveis por parte de outros profissionais de saúde que não o autor do registo, dificultando assim a partilha de informação, quando necessário (p.ex. entre outros médicos especialistas);

- Segurança de dados (persistência e disponibilidade) – Um registo de saúde eletrónico bem concebido, contempla a importância da informação que possui e potenciais falhas que possam ocorrer, sendo que contem mecanismos de recuperação de desastres, com sistemas de backup em suportes físicos independentes e

eventualmente redundância de serviço, dependendo dos requisitos da instituição de saúde a que se destina. O registo eletrónico, acaba por ser muito mais seguro do que um sistema baseado em papel, onde a redundância de dados obrigaria a problemas logísticos enormes;

- Economia de espaço – Os registos eletrónicos, apesar de serem mais completos que os seus congéneres em papel, e de eventualmente não serem tão replicados devido à necessária redundância de dados eletrónicos, estes ocupam uma fração do espaço físico, o que é compatível com o crescente aumento de registos, não sendo necessário ser feita (à partida) qualquer tipo de reciclagem de registos mais antigos;

- Confidencialidade de dados e pacientes – O acesso às fichas pode ser restringido e diferenciado (p.ex. diferentes permissões para o médico de família, especialista, enfermeiro, etc.), assim como monitorizado e registado em Logs (por questões de controlo e responsabilização de acessos);
- Flexibilidade de vistas e layout – Dependendo da implementação do EHR e das respetivas características da camada de apresentação e opções de configuração, a informação pode ser apresentada da maneira mais adequada ao leitor. Mais importante ainda, é a possibilidade de apresentar a informação selecionada e ordenada, de acordo com vistas específicas, apropriadas para diferentes consultas (p.ex. registos ordenados cronologicamente, seleção de informação sob uma certa doença ou estruturada de acordo com um determinado formato, etc.);

- Integração de vários recursos informacionais – Os registos de saúde eletrónicos podem conter não apenas as anotações de consulta dos médicos (típicas das fichas em papel) como também agregar e disponibilizar em tempo real, outros recursos informacionais, como exames e imagens correspondentes. Sejam estas guardadas localmente ou apenas referenciadas e acessíveis remotamente;

- E várias fontes de informação – A captura dos dados guardados, pode ser feita diretamente de equipamento médico especializado (como monitores cardiovasculares, equipamento de análise laboratorial, etc.), sem perdas de informação em conversões de formatos (digitalização de papel) e em tempo real (disponibilização imediata, após medição);

- Processamento contínuo de dados – A informação pode ser continua e automaticamente processada, com verificações de erros, integridade e consistência de dados, assim como interpretada no seu contexto de maneira a possibilitar alertas e avisos, aos clínicos, sobre eventos relevantes (p.ex. valores anormais de um exame);

- Pesquisa assistida e rápida – O tratamento eletrónico da informação possibilita a pesquisa e recuperação da ficha de um determinado paciente, numa fração do tempo utilizado num sistema de ficheiros físico. Além disso permite fazer pesquisas diretamente sob os dados, sejam eles estruturados, como campos concretos de uma base relacional, ou não estruturados, como palavras específicas num texto corrido, ou ainda num futuro próximo, a imagens através das suas características (com a aplicação

dos conceitos de base de dados multimédia e contentasse retirava á informática médica);

- Centralização e atualização constante da Informação – de maneira a que esta possa ser partilhada por vários prestadores, evitando não só registos contraditórios assim como duplicações de exames e consequentes gastos supérfluos. Além disso a informação é disponibilizada a outros clínicos, instantaneamente após ter sido inserida;

- Estruturação da informação – de acordo com standards médicos normalizados, de maneira a garantir o correto e completo registo de dados, e promover a interoperabilidade dos mesmos, por vários prestadores/especialidades. Além disso facilita potenciais reestruturações futuras, de acordos com novos formatos adotados;

- Controlo estatístico e otimização de prestação de serviços – O facto de os registos serem eletrónicos, permite uma recolha muito mais fácil e rápida de informação estatística e seu processamento (informação recolhida respeitando o anonimato dos pacientes), contribuindo assim para a constante otimização e melhoramento dos processos clínicos assim como para a análise global dos serviços. Os dados recolhidos podem ser usados até mesmo para fins académicos e de desenvolvimento, contribuindo assim ainda mais para o desenvolvimento da medicina.

## 2.5 E-government

Neste ponto irá ser dada uma descrição e caracterizado o conceito de e-government, de forma a poder ser considerado no enquadramento com o tema da tese.

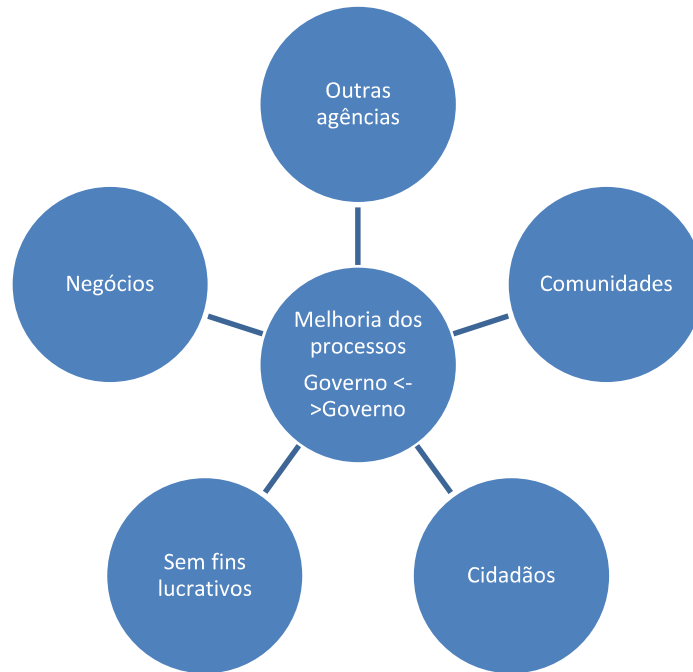
E-government é o uso de tecnologias de informação e comunicação para melhorar as atividades de organizações do setor público, neste caso específico de IPSS.

Algumas definições restringem e-government com aplicativos compatíveis com a Internet, ou apenas com interações entre grupos de fora do governo.

Existem três áreas no e-government:

- Aumento da performance dos processos governamentais;
- Conexão com os cidadãos;
- Construção de interações com o exterior.

De acordo com “eGovernment for Development Information Exchange” da Universidade de Manchester a figura 2 apresenta um esquema que caracteriza o funcionamento do e-government com o objetivo de interligar todas as valências com o intuito de melhorar os processos governamentais.



**Figura 2** - Esquema com as diferentes entidades do e-government

As iniciativas do e-government de domínio particular com o funcionamento do setor público incluem:

Cortar custos do processo - melhorar os dados de entrada em relação com a com os outputs cortando custos financeiros e de tempo;

Gestão do desempenho dos processos - planejamento, monitorização e controle de desempenho dos recursos de processo desde humanos, financeiros entre outros;

Conexões estratégicas no governo - Ligações com as agências do governo fortalecendo os dados conseguindo uma melhoria na investigação, desenvolvimento e implementando desta forma uma estratégia que oriente processos governamentais;

Criação de Poder – transferência de poder, entre a autoridade e os recursos de acordo com os processos locais existentes para novos locais.

No que toca á conexão dos cidadãos, as iniciativas tratam muito especificamente a relação entre governo e cidadãos, ou como eleitores as partes interessadas em que o setor público deve trabalhar de forma legítima fornecendo um serviço de qualidade aos cidadãos, estas iniciativas podem muito bem incorporar as melhorias dos processos identificadas nos pontos anteriores, no entanto também incluem outros pontos mais amplos, são eles:

- Comunicar com os cidadãos – proporcionar aos cidadãos detalhes da atividade do setor público, trata-se sobretudo de certos tipos de responsabilidade de forma a

obrigar os funcionários públicos a serem mais responsáveis pelas suas decisões e ações;

- Ouvir os cidadãos – aumentar a colaboração dos cidadãos nas decisões do setor público;
- Melhoria dos serviços públicos – melhorar os serviços prestados aos membros do público tais como qualidade, conveniência e custo.

Tais iniciativas tratam muito especialmente a relação entre órgãos públicos e outras instituições – outros órgãos públicos, empresas privadas, sem fins lucrativos e organizações comunitárias. Tal como acontece com as ligações com os cidadãos, estas iniciativas podem muito bem incorporar as melhorias dos processos, tais como:

- Trabalhar melhor com os negócios – melhorar a interação entre governo e as empresas, isso inclui a regulação de aquisição de serviços para e de negócios para melhorar a qualidade, conveniência e custo;
- Desenvolvimento de comunidades – desenvolvimento das capacidades sociais e económicas e do capital das comunidades locais;
- Construção de parcerias – criação de agrupamentos organizacionais para atingir objetivos económicos e sociais, o setor público é quase sempre um dos parceiros, embora ocasionalmente ele atua apenas como um facilitador para os outros.

### 3 Metodologia

Neste capítulo, apresenta-se a metodologia escolhida para recolha e análise de dados – Action Research. Serão abordados diversos aspetos relevantes desta metodologia, nomeadamente uma introdução histórica e os seus principais processos.

A análise que se apresenta em detalhe ajudará a compreender melhor as boas práticas recomendadas pela abordagem da Action Research para que, no caso de estudo, seja possível escolher aquela que melhor se enquadra para o projeto apresentado no mesmo.

Os métodos de recolha e análise de dados foram documentos existentes na instituição, análise de Software existente, entrevistas e observação direta.

### 3.1 Action Research

A metodologia escolhida foi a Action Research (AR), tendo em conta as palavras de Dick (1997) que afirmou a flexibilidade suficiente para permitir o começo impreciso, progredindo no sentido de conclusões adequadas. Seguindo os princípios básicos de AR, como descrito acima, o conhecimento gerado a partir desta mudança decreta pesquisa. Esta mudança deve-se ao desejo de contribuir em termos de conhecimento para a compreensão dos sistemas de informação utilizados no lar de idosos caso de estudo e, em geral, a oferecida pela instituição da segurança social. Por outro lado, verificou-se que a participação e a emergência de conhecimento ocorreu durante as fases de recolha de dados. No entanto, se a mudança vai ou não ser aceite em situações profissionais não é dependente dos desejos do pesquisador.

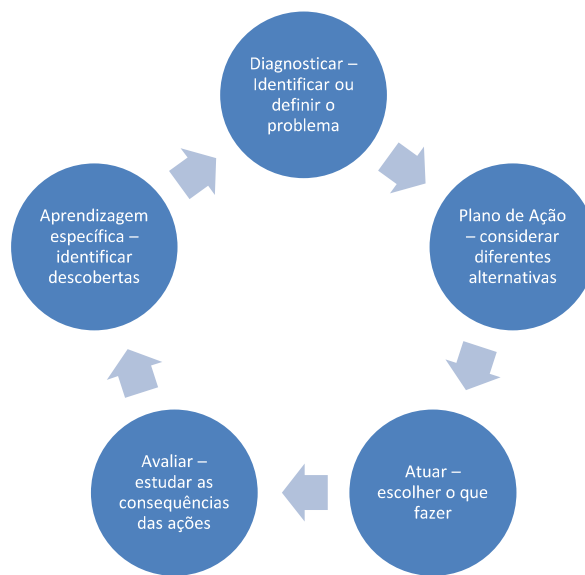
De acordo com Kemmis & McTaggart (1988:05) AR é "uma forma de investigação autorreflexiva coletiva empreendida por participantes em situações sociais, a fim de melhorar a racionalidade e a justiça de suas próprias práticas sociais ou educacionais, bem como a sua compreensão destes práticas e as situações em que estas práticas são levadas a cabo (...) "Na pesquisa-ação o envolvimento de todas as partes fornece mais informações sobre a situação. Este processo é bem adequado para situações em que desejam alcançar a mudança e compreensão, ao mesmo tempo, como é o caso na presente investigação.

AR, de acordo O'Learly (2004), é frequentemente usado em locais de trabalho onde a propriedade de mudança é uma prioridade alta ou onde o objetivo é melhorar a prática profissional. A identificação de problemas práticos em um contexto específico e as tentativas de buscar e implementar soluções dentro desse contexto foram considerados. Há alguns princípios básicos da AR considerados por O'Learly (2004:139):

- Gerar conhecimento e mudança decreta - AR é "mais do que apenas mudar a implementação e depende tanto a produção de conhecimento para produzir a mudança e a aprovação da mudança de produzir conhecimento". Isto significa que o processo de conhecimento é altamente integrado com a mudança. Por outro lado, O'Learly (2004) afirma que "se estar a desenvolver habilidades técnicas, a construção de mais reflexiva a prática profissional, ou emancipação e libertação, pesquisa-ação trabalha para a melhoria situação com base na prática, e evita o problema da necessidade de trabalhar para a mudança após conhecimento é produzido";
- AR é participativo - um princípio central da AR é a "democratização" do processo de investigação. Isto exige a participação e a colaboração entre pesquisadores, pessoal da instituição, e quaisquer outras partes interessadas;
- AR depende de um processo cíclico - AR é um processo cíclico que toma forma como o conhecimento emerge. Nesta pesquisa, os ciclos convergiram para uma melhor compreensão da situação e permitiram que as implementações melhorem a ação, pois

são baseados na prática avaliativa. O pesquisador continuamente irá refinando os métodos de dados, e interpretação à luz da compreensão desenvolvida nos ciclos anteriores.

De acordo com Rory O'Brien "An Overview of the Methodological Approach of Action Research" a figura 3 apresenta um esquema ilustrativo da metodologia de forma a representar a comunicação entre os vários estados da mesma.



**Figura 3** - Esquema das diferentes etapas da metodologia Action Research

Dentro da pesquisa-ação uma série de métodos podem ser escolhidos para a recolha de dados. As abordagens para recolher informações sobre a situação, os utilizadores, clientes, colaboradores e outras partes interessadas foram documentos institucionais, as atitudes da comunidade (enfermagem, auxiliares, médicos e instituição de segurança social) e aplicações de Software em uso. Em seguida, outros métodos foram utilizados: observação sobre as pessoas e o ambiente de trabalho e entrevistas aos colaboradores.



## 4 Caso de Estudo

Neste capítulo será apresentado um caso de estudo, com o objetivo de acompanhar a construção de um Software, onde se pretende otimizar o tempo e os recursos para a sua execução com sucesso. Devido às necessidades recentes relacionadas com a conjuntura atual da economia, a instituição CSAL necessita agilizar os seus processos, para que toda a informação, antes espalhada, seja agora concentrada numa única ferramenta, para ser alvo de tratamento mais especializado.

## 4.1 Centro Social Amigos da Lardosa (CSAL)

O centro social de amigos Lardosa (CSAL) foi o estudo de caso escolhido. O CSAL está em uma aldeia situada a cerca de 18 quilômetros da capital do distrito de Castelo Branco. Em 2007, um estudo da DECO diz que Castelo Branco foi considerada uma das melhores cidades para viver em Portugal e a 7<sup>a</sup> melhor cidade para se viver na Europa, devido à sua excelente qualidade de vida. Embora haja um cuidado domiciliar que auxilia clientes na sua própria casa, o objetivo principal do CSAL, uma instituição particular de solidariedade social, é ajudar os clientes que não conseguem ter, em suas casas, as condições de apoio para o necessário cuidado de uma boa qualidade de vida.

O CSAL oferece vários serviços diferentes, ou seja, a estrutura residencial para idosos, serviços de apoio domiciliário, centro de dia, serviços de fisioterapia e atividades de tempos livres para crianças até 10 anos de idade. A instituição tem uma cozinha para preparar as refeições a qual é também responsável por fornecer e alimentar as crianças da escola da aldeia.

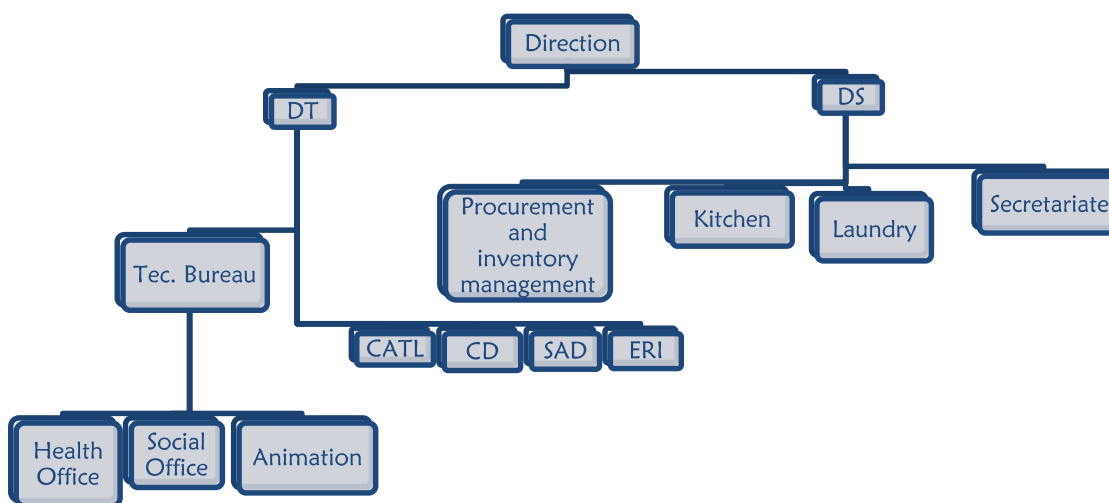


Figura 4 - Estrutura hierárquica do Centro Social Amigos da Lardosa

Na figura 4 é apresentada a estrutura hierárquica do CSAL. Dois grupos principais podem ser distinguidos, estando ambos a cargo dos diretores técnicos e serviços. A área de diretor técnico é dividida em gabinete técnico e as valências que o centro disponibiliza: enfermagem, atividades de lazer do centro (CATL) e creche (CD), serviço de apoio domiciliário (SAD) e as estruturas residenciais para idosos (ERPI). Estes serviços têm diferentes tipos de trabalhadores: os prestadores de ajuda direta e auxiliares de serviços, ao gabinete técnico reportam os serviços de saúde, com um

fisioterapeuta, um médico e um enfermeiro. O escritório social tem uma socióloga e um educador social. Na área supervisionada pelo diretor de serviço enquadram-se os stocks de gestão e o secretariado, cujas tarefas são desenvolvidas pelo diretor. A gestão da cozinha está a cargo do diretor de serviços bem como a lavandaria, onde as tarefas são realizadas por auxiliares de serviços.

Como foi referido no capítulo 3, o estudo da informação do centro foi realizado através da utilização de diferentes métodos de pesquisa. Os dados foram recolhidos a partir de entrevistas com a equipe, observações e análise de informações do sistema.

### 4.1.1 Sistemas de Informação no CSAL

Geralmente, as aplicações para gestão de sistemas de informação disponíveis no mercado são dispendiosas. Por outro lado, o Software não está sempre bem estruturado para enfrentar a personalização do cliente.

No momento, esta instituição tem dois programas de Software: um é responsável pela gestão de conta, o outro é para a gestão dos utilizadores (adquirido recentemente para ser usado como uma base de dados). No entanto, as dificuldades dos funcionários, para lidar com aplicações de informática, condicionou o aproveitamento integral do mesmo. O Software para gestão de utilizadores tem as seguintes características (ver tabela 1).

A Tabela 1 apresenta dois aspetos que podem ser resolvidos através dos utilizadores de Software de gestão. Através deste Software o utilizador pode: criar e gravar todos os parâmetros dos recibos que a instituição pode manter; e fazer os cálculos sobre o que cada cliente gasta. A indicação das pessoas que se encontram em lista de espera pode também ser gerada.

O uso do Software é extremamente complexo, o que levou a que, atualmente, a sua utilização fosse virtual, apenas para criar o registro do cliente.

**Tabela 1 - Controle de clientes vs controlo de utilizadores**

<b>Clients Control</b>	<b>Users Accounting Control</b>
Waiting list with ability to set properties	Automatic calculation (monthly)
User Registration (Documents Control)	Receipts Issue from Users
User Registration per Valence (Documents Control)	Receipts Issue from Users with Bar Codes
Family Visits Control	Payments Monitoring
Documents Attachment (Various Formats)	Maps (IRS, current account, etc.)
Users Misconduct Control	General Receipt Control

A Tabela 2 apresenta todas as características da contabilidade gerencial. Este Software só pode ser funcional para um contabilista com alguma experiência, tal como o Software anterior, devido às suas complexidades de utilização. Todas as características são muito específicas, logo, para quem não tem conhecimentos contabilísticos terá dificuldade em utilizar o Software. Este Software está dividido em

várias áreas, cada uma com características particulares e sofre atualizações constantes de acordo com a legislação. Existem também problemas de compatibilidade com os órgãos de supervisão.

Tabela 2 - Contabilidade gerencial

<b>Maps</b>	Maps balance, DRE, DRF, DRL between periods
	Memory Map estimate Justification of the operating account
<b>Balance sheets</b>	Synthetic balance and analytical parameterization
	Balance by general valence
<b>Modules</b>	Recovery module IVA
	Budget figures module
<b>Calculations</b>	Cost calculation/ prescription by user and valence
	Costs
	Income
<b>Budgets</b>	General
	For valence
	Provisional Operation
	Duodecimal Method
<b>Results</b>	Financial
	Extraordinary
<b>Movements</b>	Automatic adjustment of stocks
	Economic
	Financial
	Activity
	Banking
<b>Models</b>	Extracts
	1042
	1035
	1031
	1044
	1036
	1037
	1043
	1041
	1045
	1046
	1038
	1039
	1040
2745	
<b>Emissions</b>	Sheet 1 da DGSSFC
	Sheet 2 da DGSSFC

Os principais problemas encontrados com o uso dos sistemas de informação basearam-se na ausência de hipótese de personalização, na pouca experiência dos trabalhadores da instituição, e nas mudanças frequentes da legislação.

## 4.2 Análise de Dados

Para recolher a informação sobre as pessoas, a informação, o fluxo da mesma bem como outros dados necessários para a criação da aplicação, recorreu-se a uma estrutura baseada em Zachamn.

Tabela 3 - Arquitetura de Zachamn

	What	How	Where	Who	When	Why
<b>Scope</b>	Several information systems directed to the social sector; Description of the services rendered by social security and by other care entities	Information collection, research and contacts with users and technicians from social security; questionnaires from other institutions; observation of entities functioning	Institution of social solidarity by healthcare service rendering to elderly; center of leisure activities; physiotherapy office, home-based care services	Direct help providers, health professionals, clients, social technicians, accountant, social staff, cooks	Available always and each time that is necessary to make control registration; statistic consultation and information forward to supervisors entities	Importance of the existing institutions for clients;  Technology relevance
<b>Business</b>	To use the new technologies in support to the cares control that are given to elderly and thus to increase the prevention for, the future, minimizing errors	To observe the modus operandi of the intervenient who render customer services; and to deal in loco with users	To join together the way controls are made and to define a pattern of the executed tasks	Users that will use the platform actively, namely, technical director, service director, social staff, direct help providers, nurse, doctor, and physiotherapist	Periodical contacts to clarify all the doubts that can appear and thus to establish a user interaction; to visit other institutions in order to verify identical problems	To contribute to a better and secure organization, giving all the necessary statistics to the social security institution; warranting customers better quality
<b>Information Systems for Elderly Care</b>	Mode of Information Systems Functioning	Use case diagrams, sequence diagrams, state diagrams, architectures, security modes	Test areas in controlled environment, test patterns defined, questions from users to prevent errors	Different access forms of the institution collaborators	Definition of a compulsory cycle to take and analyze the statistical information	To define each information system function, to redefine the mode of functioning, to define procedures
<b>Technology</b>	Programming Languages, protocols and	To consult the current practical procedures;	Institution Installation, existing network	Organization of health services providers to	To establish deadlines for	Use of information systems

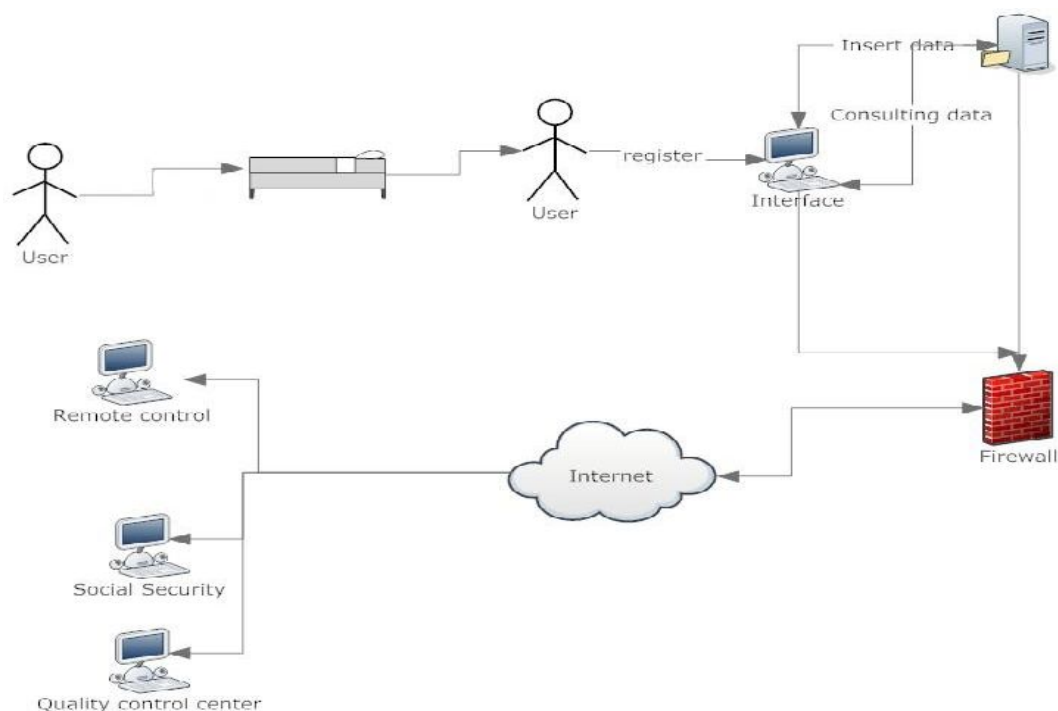
other existing systems; compulsory patterns of information rendering	quality manuals from social security	architecture, and in remote environment, link with the system	elderly, institute of social security, social staff	frequent tests; possibility for 24 hours tests	transparency, new methods of registration and healthcare providers, information treatment, archive, information privacy
--	--------------------------------------	---	---	--	---

O objetivo principal do quadro de recolha de dados foi definir as necessidades específicas e os problemas encontrados da instituição. Após a primeira imagem da instituição, os dados começaram a ser organizados e as ideias surgiram para o desenho do quadro final. A razão para a escolha do modelo de Zachman (1987) foi a necessidade de criar e classificar as ferramentas e documentos, a fim de ordenar os problemas complexos no domínio dos sistemas de informação.

A estrutura Zachman para arquitetura corporativa é um esquema de classificação seis por seis, onde as seis linhas representam diferentes perspetivas da empresa e as seis colunas ilustram diferentes aspetos. Para garantir uma compreensão completa e holística da arquitetura corporativa, é necessário desenvolver modelos que atendam às perspetivas e aos aspetos que constituem as linhas e colunas do quadro, respetivamente.

Após essa etapa, diversas entrevistas semiestruturadas foram feitas. O objetivo era compreender as dificuldades que o pessoal, em asilos, têm em lidar com toda a quantidade de informação que está espalhada por toda a instituição. Desde o primeiro estágio da análise dos dados pode ser desenhado um quadro que deve estar preparado para ser uma diretriz para as diferentes instituições de solidariedade social (IPSS) na troca dos diversos tipos de comunicação.

Em termos de esboço, a arquitetura do sistema para a interação referida, pode ser proposta como a apresentada na figura 5. O projeto de arquitetura do sistema permitirá alinhar várias visões da mesma realidade, e compreender as suas forças e fraquezas.



**Figura 5** - Esquema de funcionamento do Software existente

Atualmente, a arquitetura começa a ser melhorada e para expressar a nível global os elementos básicos da arquitetura corporativa.

Teoricamente, uma vez que estamos em fase de pesquisa, os resultados que esperamos com o framework de desenvolvimento e implementação são contribuir para melhorar a gestão da instituição, tanto a nível da burocracia como prático.

Primeiro que tudo, as informações poderiam ser centralizadas numa única plataforma com todos os campos discriminados, deste modo, as necessidades podem ser encontradas num só local.

Outra situação que encontramos é a ausência de privacidade e ética no processo de consulta de documentos, uma vez que toda a informação (contabilidade, serviço de atendimento) em formato de papel está de livre acesso. O quadro irá incluir essa restrição de uso, de forma a permitir que os utilizadores apenas possam aceder á informação que lhes compete.

O objetivo deste quadro servirá aos diretores técnicos facilmente realizar a extração de informações da instituição que não era possível na área de contabilidade, onde eles já têm algumas ferramentas que podem usar para esses mesmos mapas. Na área de gestão de utilizadores, especialmente considerando os serviços de cuidados prestados, eles só tinham alguns relatórios gerais que eram difíceis de encontrar e de tê-los atualizados.

Este quadro também dará a possibilidade de refletir sobre o impacto ambiental. Portanto, nessas instituições o uso de papel é uma realidade. O quadro poderá contribuir para atenuar alguns danos através da informática para utilizar processos de troca, relatórios, pontos de situação, e assim o papel irá ser reduzido para o indispensável, e a instituição irá tornar-se livre de papel.

## **5 Desenvolvimento tecnológico**

Existem uma aplicação que realiza algumas tarefas idênticas à aplicação desenvolvida, no entanto não permite fazer exatamente o mesmo. Podem realizar algumas tarefas diferentes, mas não abrangem todos os objetivos pretendidos. Este capítulo aborda essa mesma aplicação.

## 5.1 F3M Information Systems, S.A (F3M)

A empresa escolhida foi proposta pelos colaboradores do centro, uma vez que se trata de uma empresa em que o core business é exatamente o desenvolvimento de software para este sector e a que é largamente utilizada nos outros centros.

A F3M Information Systems, S.A. é uma das maiores empresas portuguesas especializadas em Tecnologias da Informação e Comunicação.

Fundada em julho de 1987, tem sede em Braga e escritórios em Lisboa, Luanda e Maputo. A F3M Information Systems, SA assume-se com uma fortíssima vertente de conceção, produção e implementação de Software para mercados verticais, vertente essa que coexiste com uma forte atividade nas áreas de fornecimento de soluções TIC, como infraestrutura tecnológica, telecomunicações, Software de gestão e serviços de consultoria.

Ao longo de quase três décadas empresa foi acumulando um vasto conjunto de certificações e distinções, destacando-se nos últimos anos as seguintes:

- Certificação em Investigação, Desenvolvimento e Inovação (NP 4457), sendo uma das poucas dezenas de empresas portuguesas que obteve esta certificação;
- Integrante da Rede PME's Inovadoras COTEC;
- Caso de estudo na área de Responsabilidade Social das Organizações;
- Considerada uma das 100 melhores empresas para trabalhar pelo quinto ano consecutivo (desde 2011 - Exame/Accenture).

A F3M assume uma posição de destacada liderança no mercado nacional de fornecimento de Software para a área da Economia Social. Uma liderança assente em mais de duas décadas de existência com incontestável sucesso, na confiança das 3000 instituições clientes, tendo desenvolvido um acordo com a Confederação Nacional das Instituições de Solidariedade e sido selecionada como fornecedor da União das Misericórdias Portuguesas.

Para o setor social a F3M disponibiliza um vasto leque de programas que asseguram uma total cobertura das necessidades ao nível da gestão e organização das entidades do setor.

Acompanhando sempre as exigências legais e funcionais, as soluções F3M apresentam características únicas no panorama nacional de Software para o setor social. O nível de integração entre as soluções F3M constitui uma importante mais-valia para os utilizadores que evitam assim, tarefas de dupla introdução de dados.

Segue-se um esquema na figura 6 de como o Software está interligado.

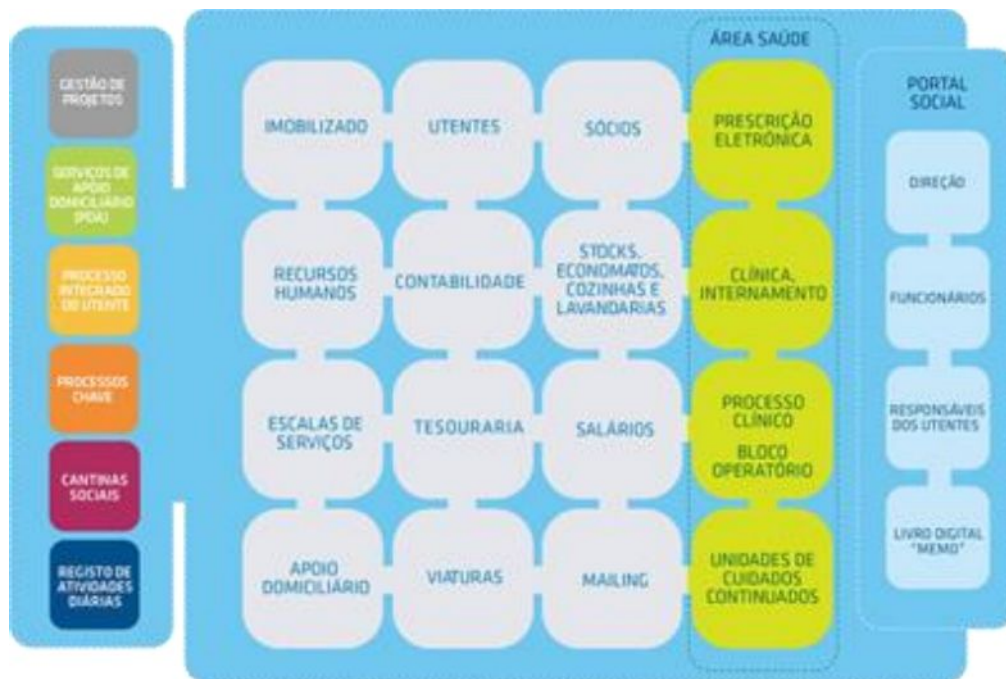


Figura 6 - Esquema de Software F3M

## 5.2 Tecnologias utilizadas

A escolha das tecnologias utilizadas baseou-se no facto de estarem bastante difundidas e algumas delas serem líderes dentro da sua categoria.

Todas as tecnologias utilizadas no desenvolvimento do projeto são versões express, por serem de código aberto proporcionando um contributo à escala global, e como a licença é livre de custos torna-se bastante acessível a qualquer pessoa que queira continuar o desenvolvimento desta aplicação. Vão ser explicadas e referidas, no geral, as razões da escolha das tecnologias selecionadas. Será feita uma descrição técnica de cada uma delas.

### 5.2.1 Internet Information Services (IIS)

Anteriormente denominado de Internet Information Server, é um servidor web criado pela Microsoft para os seus sistemas operacionais. A sua primeira versão foi introduzida através do Windows NT Server versão 4, e passou por várias atualizações. É também o sucessor do PWS (Personal Web Server). A versão mais recente é o IIS 8.5 (disponível apenas no Windows Server 2012 R2 e Windows 8.1). A função do IIS no Windows Server é oferecer uma plataforma para o alojamento de sites, serviços e aplicativos, sendo capaz de integrar as seguintes tecnologias: ASP.NET, FTP, PHP, WCF e o próprio IIS.

Uma das características mais utilizadas é a gestão de páginas HTML dinâmicas, que ao contrário de outros servidores web, usa tecnologia proprietária, o ASP (Active Server Pages), mas também pode usar outras tecnologias com a instalação de outros módulos.

Para utilizar essa ferramenta é necessário adquirir uma licença de uso que para cada instalação ou versão é preciso efetuar o devido pagamento.

Depois do lançamento da Plataforma .NET em 2002 o IIS ganhou também a função de gerir o ASP.NET. Este é formado basicamente por dois tipos de aplicações:

Páginas Web: Tradicionais acedidas por utilizadores, contém a extensão ASPX

Web Services: Funções disponibilizadas pela rede, chamada por aplicativos ASMX

A figura 7 apresenta de forma ilustrativa o funcionamento do IIS.

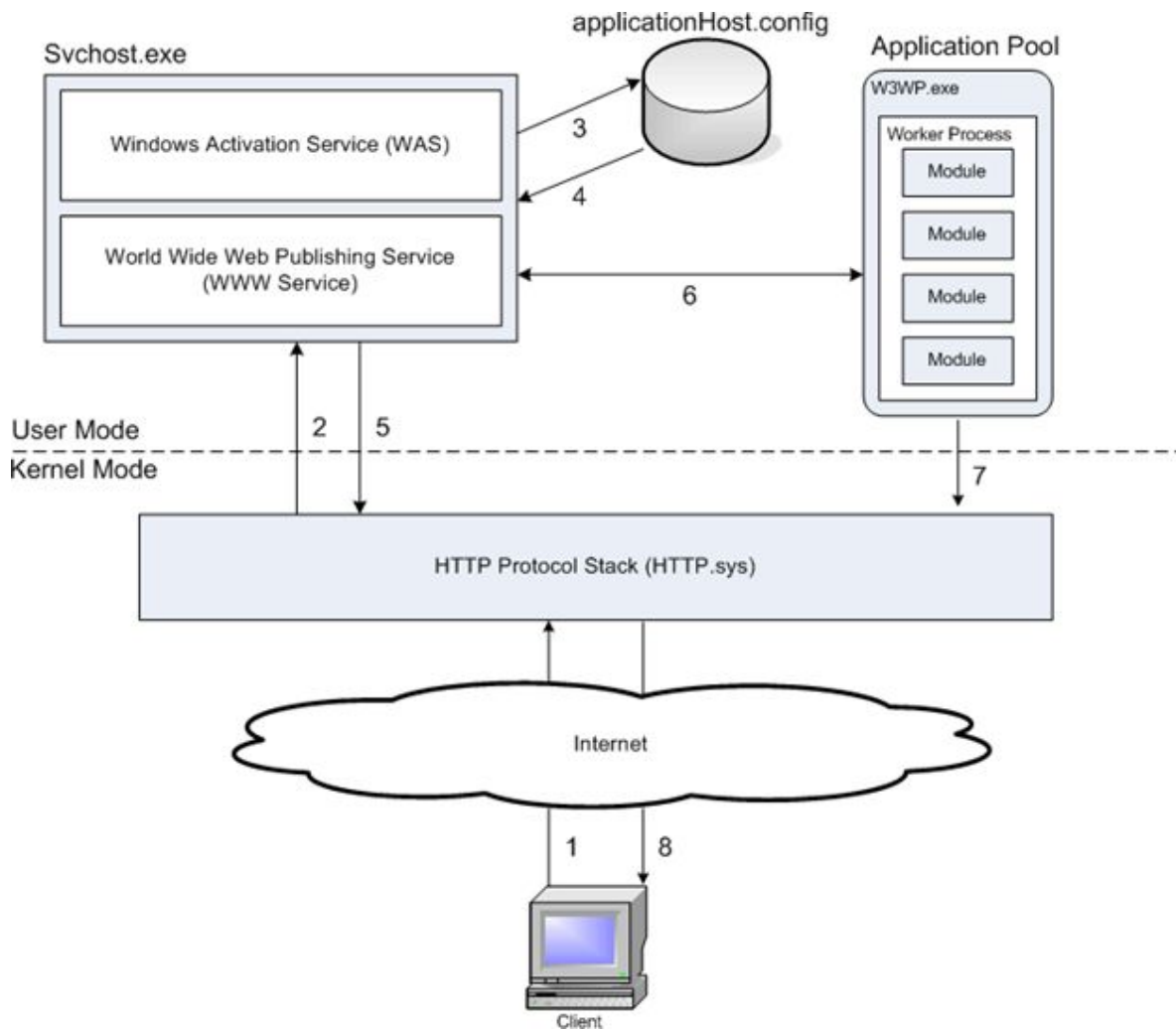


Figura 7 - Esquema ilustrativo de funcionamento do IIS.

### 5.1.2 SQL Server

O SQL Server é um sistema de gestão de base de dados relacionais (RDBMS) da Microsoft, é projetado para o ambiente corporativo. SQL Server é executado em T-SQL (Transact-SQL), com um conjunto de extensões de programação de Sybase e Microsoft que adicionam vários recursos para o SQL padrão, incluindo controlo de transações, exceção, tratamento de erros, processamento de filas e declaração de variáveis.

Todas as operações que podem ser invocadas no SQL Server são comunicadas através de um formato de Microsoft definido, chamado de Tabular Data Stream (TDS). O TDS é um protocolo da camada de aplicação, usada para transferir dados entre um servidor e um cliente da base de dados. Inicialmente projetado e desenvolvido pela Sybase Inc para o motor de base de dados relacional Sybase SQL Server em 1984, e mais tarde pela Microsoft no Microsoft SQL Server, os pacotes de TDS podem ser incorporados em outros protocolos dependentes de transporte físico, incluindo TCP / IP, named pipes e memória partilhada. Consequentemente, o acesso ao SQL Server está disponível sobre esses protocolos. Além disso, a API do SQL Server também está disponível em web services.

O SQL Server oferece suporte a diferentes tipos de dados, incluindo tipos primários, tais como Integer, Float, Decimal, Char (incluindo sequência de caracteres), Varchar (sequência de caracteres de comprimento variável), binário (para blobs não estruturados de dados), Texto (para dados textuais), entre outros. O arredondamento de tipos float para inteiros usa o Symmetric Arithmetic Rounding ou Symmetric Round Down (Fix) dependendo dos argumentos: `SELECT Round (2.5, 0)`.

Microsoft SQL Server faz as estatísticas do servidor disponíveis como tabelas virtuais e pontos de vista (chamadas de exibições de gestão dinâmica ou DMVs). Além de tabelas, uma base de dados também pode conter outros objetos, incluindo visualizações, procedimentos armazenados, índices e restrições, juntamente com um log de transações. Uma base de dados SQL Server pode conter um máximo de 231 objetos, e pode abranger vários arquivos de nível de sistema operacional com um tamanho máximo de 1 exabyte. Os dados na base de dados são armazenados em arquivos de dados primários com uma extensão mdf os arquivos de dados secundários, identificados com uma extensão ndf, são usados para permitir que os dados de uma única base de dados possam ser distribuídos por mais que um arquivo, e, opcionalmente, em mais de um sistema de arquivos. Os arquivos de log são identificados com a extensão ldf.

O espaço de armazenamento atribuído a uma base de dados é dividido em páginas numeradas sequencialmente, cada uma com 8 KB de tamanho. Uma página é a unidade básica de I / O para operações de SQL Server é marcada com um cabeçalho de 96 bytes que armazena meta dados sobre a página, incluindo o número da página, tipo de página, espaço livre e o ID do objeto que possui. O Tipo de página define os dados contidos na página: dados armazenados na base de dados, índice, mapa de alocação

que contém informações sobre como as páginas são alocados por tabelas e índices, a mudança de mapa que contém as informações sobre as alterações feitas em outras páginas desde o último backup ou Logs que contém grandes tipos de dados, como imagens. Enquanto a página é a unidade básica de uma operação de I / O, o espaço é realmente conseguido em termos de uma medida que consiste de 8 páginas. Um objeto da base de dados pode abranger todas as 8 páginas num só ponto ("extensão uniforme") ou compartilhar uma extensão com até 7 mais objetos ("extensão mista"). Uma linha numa tabela da base de dados não pode abranger mais que uma página, por isso é limitado a 8 KB de tamanho. No entanto, se os dados excederem 8 KB e a linha conter dados VARCHAR ou Varbinary, os dados nessas colunas são movidas para uma nova página (ou, possivelmente, uma sequência de páginas, chamada de unidade de alocação) e substituído por um ponteiro para os dados.

Para armazenamento físico de uma tabela, as linhas são divididas em séries de divisórias (numeradas de 1 a N). O tamanho da partição é definida pelo utilizador, por padrão, todas as linhas estão numa única partição. Uma tabela é dividida em várias partições, a fim de espalhar a base de dados ao longo de um cluster de computadores. As linhas em cada partição são armazenadas em qualquer árvore ou estrutura de heap. Se a tabela tem, um índice de cluster associado para permitir a recuperação rápida de linhas, as linhas são armazenadas em ordem de acordo com os seus valores de índice, com uma árvore fornecendo o índice. Os dados estão no nó entre folhas, e os outros nós que armazenam os valores de índice para a folha de dados conseguir ficar acessível a partir dos respetivos nós. Se o índice for não-agrupado, as linhas não são classificadas de acordo com as chaves de índice. Uma exibição indexada tem a mesma estrutura de armazenamento que uma tabela indexada, uma tabela sem um índice de cluster é armazenado numa estrutura de pilha desordenada. No entanto, a tabela pode ter índices não agrupados para permitir a recuperação rápida de linhas, em algumas situações a estrutura heap tem vantagens de desempenho sobre a estrutura de cluster.

### 5.1.3 Crystal Reports

Crystal Reports é uma aplicação de business intelligence, atualmente comercializada para as pequenas empresas, através da SAP. É usado para projetar e gerar relatórios a partir de uma ampla gama de bases de dados.

Crystal Reports permite aos utilizadores projetar graficamente uma conexão de dados e layouts do relatório. No Database Expert, os utilizadores podem selecionar tabelas de uma ampla variedade de base de dados, incluindo documentos do Microsoft Excel, Oracle, Microsoft SQL Server, Microsoft Access, pontos de vista de negócios através do Business Objects Enterprise e informações do sistema de arquivos locais.

Para gerar os relatórios podem ser colocados campos a partir destas fontes na área de design do relatório, e também podem implementá-los através de fórmulas personalizadas (usando BASIC ou a própria sintaxe do Crystal Reports), que são então colocadas na área de design. As fórmulas podem ser avaliadas em várias fases durante a implementação do relatório, conforme especificado pelo programador.

Ambos os campos e fórmulas têm uma grande variedade de opções de formatação disponíveis, que os designers podem aplicar absolutamente ou condicionalmente. Os dados podem ser agrupadas em bandas, suporta sub-relatórios, gráficos, e uma quantidade limitada de funcionalidade de GIS (Geographic information system).

#### 5.1.4 ASP.NET

ASP.NET é uma framework open-Source de aplicação Web do lado do servidor projetado para o desenvolvimento Web, foi desenvolvido pela Microsoft para permitir que os programadores possam construir sites dinâmicos, aplicações web e web services.

Foi lançado em Janeiro de 2002 com a versão 1.0 do .NET Framework, e é o sucessor do Active Server Pages, a tecnologia da Microsoft (ASP). ASP.NET é construído sobre a Common Language Runtime (CLR), permitindo que os programadores escrevam código ASP.NET usando qualquer linguagem .NET suportada. O quadro de extensão SOAP ASP.NET permite que os componentes do ASP.NET possam processar mensagens SOAP.

ASP.NET está num processo de ser reimplementado como framework web moderna e modular, juntamente com outros, como Entity Framework. O novo quadro vai fazer uso do novo open-Source .NET Compiler Platform (nome de código "Roslyn") e multiplataforma. ASP.NET MVC, ASP.NET Web API, as Páginas Web ASP.NET irão se fundir em apenas um MVC unificado. O projeto é chamado de "ASP.NET vNext".

Páginas da Web ASP.NET, conhecidas oficialmente como Web Forms, são os principais blocos de construção para o desenvolvimento de aplicações em ASP.NET. Existem duas metodologias básicas para Web Forms, um formato de aplicação web e em formato web site. Aplicações Web precisam ser compiladas antes da implementação, enquanto que os web sites permitem ao utilizador copiar os arquivos diretamente para o servidor sem compilação anterior. Os Forms web estão contidos em arquivos com a extensão ".aspx", esses arquivos contêm tipicamente conteúdo estático (X) HTML.

Pode incluir controlos de Web do lado do servidor e controlos de utilizador que foram definidos no quadro ou na página web. Por exemplo, existe um componente de caixa de texto que pode ser definido numa página como `<asp:textbox id='myid' runat='server'>` que será colocado numa caixa de entrada HTML. Além disso, o código dinâmico, que é executado no servidor, pode ser colocado numa página dentro de um bloco `<% -- dynamic code -- %>`, que é semelhante a outras tecnologias de desenvolvimento Web como PHP, JSP e ASP.

Com o ASP.NET Framework 2.0, a Microsoft introduziu um novo modelo de código subjacente que permite que o texto estático possa permanecer na página aspx, enquanto o código dinâmico permanece num arquivo aspx.vb, aspx.cs ou aspx.fs (dependendo da linguagem de programação usada). A figura 8 apresenta o esquema ilustrativo do funcionamento do ASP.net.

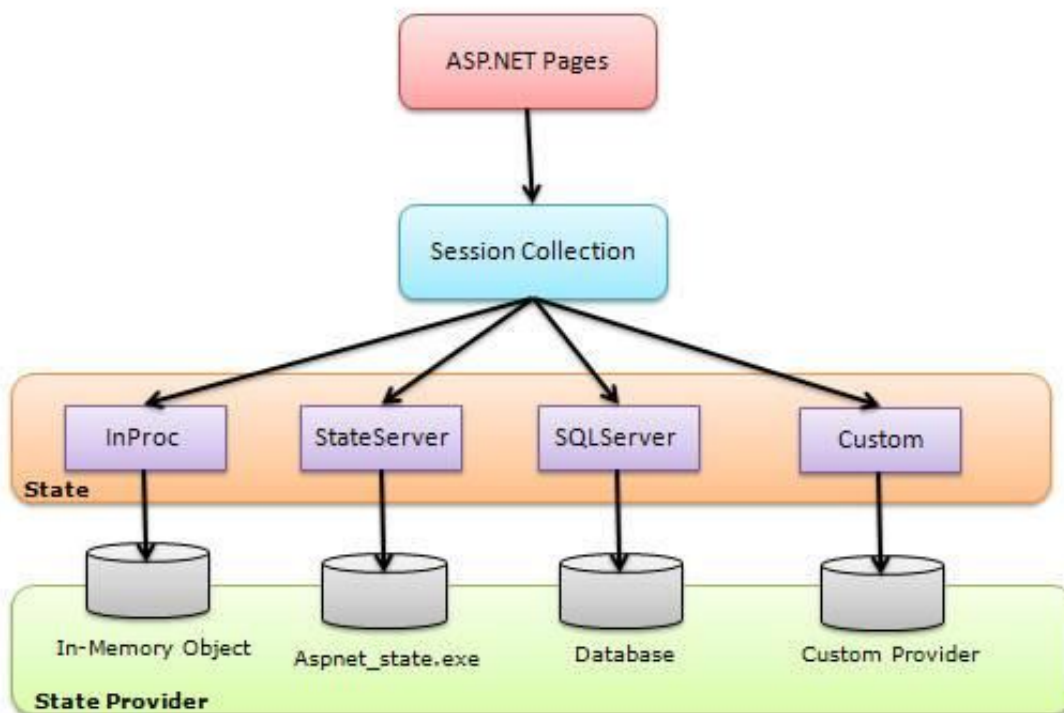


Figura 8 - Esquema ilustrativo do funcionamento do ASP.net

### 5.1.5 Hyper Text Markup Language (HTML)

Hyper Text Markup Language, comumente referido como HTML, é a linguagem de marcação padrão usada para criar páginas web. Juntamente com CSS e JavaScript, HTML é uma tecnologia fundamental, utilizado pela maioria dos sites para criar páginas web visualmente atraentes, interfaces de utilizador para aplicações web, e interfaces de utilizador para muitas aplicações móveis. Os browsers da Web podem ler arquivos HTML e torná-los em páginas visíveis.

HTML descreve a estrutura de um site semanticamente junto com sugestões para apresentação, tornando-se uma linguagem de marcação, em vez de uma linguagem de programação.

Elementos de HTML formam os blocos de construção de todos os sites HTML permitindo que as imagens e os objetos a ser incorporados possam ser utilizados para criar as formas interativas, fornecendo um meio para criar documentos estruturados por semântica estrutural para o texto como cabeçalhos, parágrafos, listas, links, citações e outros itens.

A linguagem é escrita na forma de elementos HTML que consistem em marcas entre colchetes angulares (como <HTML>). Os Browsers não exibem as tags HTML e scripts, mas usam-nos para interpretar o conteúdo da página.

HTML pode incorporar scripts escritos em linguagens como JavaScript que afetam o comportamento das páginas web HTML. Os Browsers da Web também podem se referir a Cascading Style Sheets (CSS) para definir a aparência e o layout de textos entre outros.

A marcação de HTML é composta por vários componentes-chave, incluindo os chamados de tags, tipos de dados baseados em caracteres, referências de caracteres e referências de entidade. As tags de HTML mais comumente vêm em pares como <h1> e </ h1>, embora alguns elementos sejam representados vazios e por isso são não pareados, por exemplo <img>. A primeira tag em tal par é a marca inicial, e o segundo é a marca de fim (eles também são chamados de tags de abertura e fecho).

Outro componente importante da declaração de tipo HTMLDocument, que desencadeia renderização de modo de padrões.

O seguinte é um exemplo do clássico programa Olá mundo, um teste comum utilizado para a comparação de linguagens de programação, linguagens de script e linguagens de marcação. Este exemplo é utilizado usando 9 linhas de código:

```
<!DOCTYPE html>  
<html>  
  <head>
```

```
<title>This is a title</title>
</head>
<body>
  <p>Hello world!</p>
</body>
</html>
```

O texto entre `<html>` e `</html>` descreve a página web, e o texto entre `<body>` e `</body>` é a página de conteúdo visível. O texto de marcação `<title> This is a title </title>` define o título da página do browser.

A Declaração do Tipo de Documento `<!DOCTYPE html>` é para HTML5, se a declaração não for incluída, vários navegadores irão reverter para o "modo quirks" para renderização.

Os documentos HTML implicam uma estrutura de elementos HTML juntas. Estas são indicadas no documento de tags HTML, entre parênteses em ângulo p.ex.: `<p>` .

No caso simples, em geral, a extensão de um elemento é indicada por um par de tags: a "start tag" `<p>` e "tag end" `</p>`. O conteúdo do texto do elemento, se for o caso, é colocado entre estas etiquetas.

As etiquetas podem também incluir mais markup tag entre o início e o fim, incluindo uma mistura de tags e texto. Isso indica outros elementos, como filhos do elemento pai.

A marca de início também pode incluir atributos dentro da tag. Estes indicam outras informações, tais como identificadores de seções dentro do documento, identificadores utilizados para vincular informações de estilo para a apresentação do documento, e para algumas marcas, como o `<img>` usado para incorporar imagens, em referência ao recurso de imagem.

Alguns elementos, como a quebra de linha `<br>`, não permite qualquer conteúdo incorporado, texto ou outras tags, estes requerem apenas uma única tag vazia (semelhante a uma marca de início) e não usa uma marca de fim.

Muitas etiquetas, em particular a marca de fim para o elemento muito comumente usado `<p>`, são opcionais. Um browser HTML ou outro browser pode inferir o encerramento para o final de um elemento do contexto e as regras estruturais definidas pelo padrão HTML. Estas regras são complexas e não são amplamente compreendidas pela maioria dos programadores HTML.

A forma geral de um elemento HTML, é: `<tag attribute1="value1" attribute2="value2"> conteúdo </tag>`. Alguns elementos HTML são definidos como elementos vazios e assumem a forma de `<tag attribute1="value1" attribute2="value2">`. Os elementos vazios podem não incluir conteúdo, por exemplo,

a marca ou a etiqueta em linha <img> <br>. O nome de um elemento HTML é o nome usado nas tags. É de Notar que o nome da marca final é precedida por um caractere de barra, "/" e que, em elementos vazios a marca de fim não é necessário nem permitido.

### 5.1.6 Lightswitch

Visual Studio LightSwitch 2011 é uma ferramenta de desenvolvimento que ajuda a criar aplicações rapidamente. Lightswitch fornece um ambiente de desenvolvimento simplificado que permite concentrar-se na lógica de negócios, em vez da infraestrutura das aplicações.

A maioria das aplicações de negócios são aplicações do tipo Forms-over-data que fornecem uma interface de utilizador para visualizar, adicionar e/ou modificar dados. Quando se usa outras ferramentas de desenvolvimento para construir formas de de aplicações de dados, grande parte do seu tempo é gasto em tarefas repetitivas. É necessário escrever código para interagir com uma base de dados, para a interface de utilizador, e para a lógica de negócio. Quando é utilizado o LightSwitch, muito do trabalho repetitivo é feito automaticamente, na verdade, é possível criar uma aplicação de LightSwitch sem escrever qualquer código em tudo! Para a maioria das aplicações, o único código que tem que se escrever é o código que só o programador pode escrever: a lógica do negócio.

Aplicações de negócios modernas exigem muitos recursos, tais como capacidades de pesquisa, a capacidade de classificar e reorganizar tabelas, e a capacidade de exportar dados. Aplicações do tipo LightSwitch tem essas características, e mais, já estão construídas no seu interior. Além disso, as operações típicas de dados, tais como adicionar, atualizar, guardar e eliminar também são construídos no interior, como por exemplo a lógica de validação de dados básicos.

Usando os recursos de extensões no LightSwitch, é possível alterar a aparência das aplicações através da aplicação de temas, usando controlos personalizados, e usando extensões de Shell para mudar o layout. Podem ser usados tipos de negócios personalizados para reduzir a quantidade de código que se escreve e para simplificar a formatação com a interface do utilizador.

LightSwitch simplifica o desenvolvimento de aplicações de negócios com entidades de dados e ecrãs.

As entidades de dados ou tabelas, são como o LightSwitch representa os dados. É possível criar entidades de dados usando o built-in de base de dados da aplicação, ou importando dados de uma base de dados externa, uma lista do SharePoint, ou outra fonte de dados. Pode-se criar relações entre entidades, mesmo quando as entidades são provenientes de diferentes fontes de dados, podem ainda ser criadas consultas sobre os dados usando uma interface gráfica, e pode-se assim desta forma modificar ainda mais as consultas no código. Os ecrãs ou Forms, são a forma como o LightSwitch exhibe os dados, os ecrãs são baseados em modelos pré-definidos, tudo o que se tem de fazer para vincular dados a um ecrã é especificar as entidades ou consultas a serem exibidas. Depois de criar um ecrã, pode-se modificar a aparência no interface gráfico sem nenhum código necessário.

É possível lidar com a validação básica no IDE usando campos obrigatórios e comprimentos de strings. Para a validação mais complexa baseada em lógica de negócios, é necessário escrever código. Em tempos de execução, a interface do utilizador para lidar com a validação é construída nos vários diferentes ecrãs.

Para testar a aplicação, basta executá-la, pode-se fazer alterações na interface do utilizador diretamente na aplicação em execução. Por representar um papel sob permissões de depuração, pode-se testar a autenticação e as diversas permissões. Quando a aplicação estiver completa, pode implementar-se usando o ClickOnce, que permite atualizações silenciosas no futuro. A figura 9 mostra como funciona o Lightswitch.

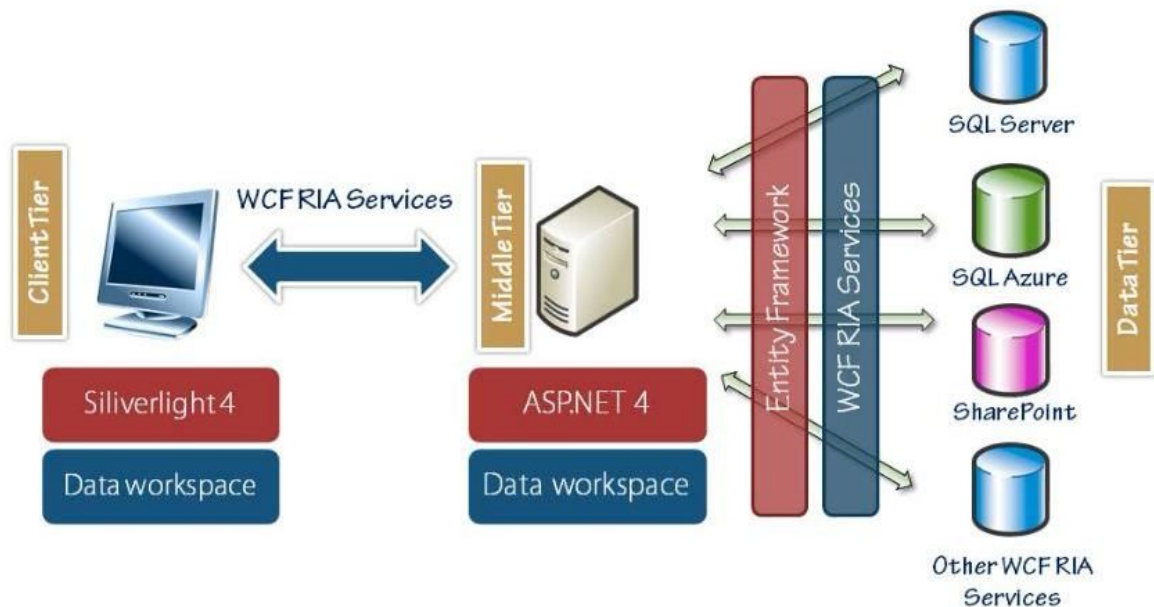


Figura 9 - Esquema de funcionamento do Lightswitch

### 5.1.7 A Linguagem de Programação C#

C# é uma linguagem de programação de multi-paradigm abrangente com uma característica forte, imperativa, declarativa, funcional, genérica e orientada a objetos disciplinados (baseada em classe) e programação orientada a objetos. Foi desenvolvida pela Microsoft no âmbito da sua iniciativa .NET e posteriormente aprovada como padrão pela Ecma (ECMA-334) e ISO (ISO / IEC 23270: 2006). C# é uma das linguagens de programação projetadas para o Common Language Infrastructure.

C# destina-se a ser de uso geral, com uma linguagem simples. Através do design, C# é a linguagem de programação que mais diretamente reflete a Common Language Infrastructure (CLI) subjacente. A maioria dos seus tipos intrínsecos correspondem ao valor de tipos implementadas pelo quadro CLI. No entanto, a especificação da linguagem não indica os requisitos de desenvolvimento de código do compilador: isto é, ele não indica que um compilador C# deve ter como alvo um Common Language Runtime, ou gerar Common Intermediate Language (CIL), ou ainda gerar qualquer outro formato específico. Teoricamente, um compilador C# pode gerar código de máquina como os compiladores tradicionais de C++ ou o Fortran.

C# suporta declarações de variáveis implícitas com a palavra-chave `var`, e implicitamente matrizes com a palavra-chave `New []` seguindo-se de um inicializador de cadeia.

C# suporta um tipo de dados booleano, as demonstrações de condições, tais como o tempo, requerem a expressão de um tipo que implementa o verdadeiro operador, tal como o tipo booleano. Enquanto C++ tem um tipo booleano, ele pode ser convertido livremente de e para números inteiros, expressões, "tais como", se requerem apenas que seja conversível por um booleano, permitindo ser um inteiro, ou um ponteiro. C# não permite esse "significado de verdadeiro ou falso", com o fundamento de que pode forçar os programadores a usar expressões que retornam exatamente `bool`, podendo prevenir certos tipos de erros de programação, tais como `if (a = b)` (uso de alocação = em vez de igualdade ==, que, embora não seja um erro em C ou C++, será considerado pelo compilador de qualquer maneira).

C# é mais seguro do que C++. As únicas conversões implícitas por defeito são aquelas que são consideradas seguras, como ampliação de inteiros, esta é imposta em tempo de compilação, durante o JIT, e, em alguns casos, em tempo de execução. Não há conversões implícitas entre Booleans e inteiros, nem entre os membros de enumeração e números inteiros (exceto para o 0, o que pode ser implicitamente convertido para qualquer tipo enumerado). Qualquer conversão definida pelo utilizador deve ser explicitamente marcada como explícita ou implícita, ao contrário do C++ que tem construtores de cópia e operadores de conversão, que são ambos implícitos por padrão.

C# tem apoio explícito da covariância e da contra variância em tipos genéricos, ao contrário do C++ que tem um grau de suporte para contra variância simplesmente através da semântica de tipos de retorno sobre os métodos virtuais.

Os membros de enumeração são colocados no seu próprio scope. A linguagem C# não permite variáveis ou funções globais. Todos os métodos e membros devem ser declarados dentro de classes, já os membros estáticos de classes públicas podem substituir as variáveis e funções globais. As variáveis locais não podem ser sombras de variáveis do bloco de inclusão, ao contrário do C e C++.

Como o C++, e ao contrário do Java, os programadores de C# devem usar a palavra-chave virtual para permitir métodos de substituição por subclasses.

Os métodos de extensão em C# permitem que os programadores usem métodos estáticos como se fossem métodos da tabela de uma classe, permitindo que os programadores adicionem métodos a um objeto que sintam que deve existir no objeto.

O tipo de dinâmica permite a ligação de método de tempo de execução, permitindo que o JavaScript execute chamadas de métodos e composições de objetos.

### 5.1.8 Visual Basic (VB)

Visual Basic é uma linguagem de terceira geração “event-driven” e com um ambiente de desenvolvimento integrado (IDE) da Microsoft para o seu modelo de programação COM. O Microsoft Visual Basic destina-se a ser relativamente fácil de aprender e utilizar. Visual Basic teve origem no BASIC e permite o desenvolvimento rápido de aplicações (RAD) de interface gráfica de utilizador (GUI), o acesso a base de dados usa Data Access Objects, objetos de dados remotos, ou o ActiveX Data Objects, criação de ActiveX e de objetos.

Um programador pode criar uma aplicação usando os componentes fornecidos pelo programa Visual Basic dentro de si. Ao longo do tempo a comunidade de programadores desenvolveu componentes de terceiros, programas escritos em Visual Basic que também podem usar a API do Windows, o que exige a declaração de funções externas.

A versão final foi a versão 6 em 1998 (agora conhecido simplesmente como Visual Basic). Em 08 de abril de 2008 a Microsoft parou de dar suporte ao Visual Basic 6.0 IDE. A equipa do Microsoft Visual Basic ainda mantém a compatibilidade para aplicações de Visual Basic 6.0 no Windows Vista, Windows Server 2008, incluindo R2, Windows 7, Windows 8, Windows 8.1, Windows Server 2012 e Windows 10 através do seu programa "It Just Works". Em 2014, havia dezenas de milhares de programadores que ainda preferiam o Visual Basic 6.0 em vez do Visual Basic.NET. Em 2014 alguns programadores pressionaram de forma a existir uma nova versão do Visual Basic 6.0.

Um dialeto do Visual Basic, Visual Basic for Applications (VBA), é usado como uma linguagem de macro ou scripting dentro de vários aplicativos da Microsoft, incluindo o Microsoft Office.

Como a linguagem de programação BASIC, o Visual Basic foi projetado para acomodar uma curva de aprendizagem. Os programadores podem criar aplicações GUI simples e complexas. A programação em VB é uma combinação de organizar visualmente componentes ou controles num só Form, especificando atributos e ações para os componentes, escrever linhas de código adicionais para maiores funcionalidades. Desde que o VB define os atributos e ações padrão para os componentes, um programador pode desenvolver um programa simples, sem escrever muito código. Programas criados com versões anteriores sofrem problemas de desempenho, mas os computadores mais rápidos e com compilação de código nativa fez este problema ficar resolvido.

Embora os programas de VB podem ser compilados em executáveis de código nativo a partir da versão 5 em diante, eles ainda requerem a presença de cerca de 1 MB de bibliotecas de tempo de execução. Bibliotecas de execução são incluídas por padrão no Windows 2000 e posterior, mas os componentes de tempo de execução, ainda têm

que ser instaladas. Versões anteriores do Windows (95/98 / NT), exigem que as bibliotecas de tempo de execução sejam distribuídas com o executável.

Os Forms são criados usando técnicas de arrastar-e-soltar. As ferramentas são usadas para colocar controles (por exemplo, caixas de texto, botões, etc.) no Form (janela), os controles têm atributos e manipuladores de eventos associados a eles. Os valores padrão são fornecidos quando o controle é criado, mas pode ser alterado pelo programador. Muitos valores de atributos podem ser modificados durante a execução, com base nas ações do utilizador ou mudanças do ambiente, proporcionando uma aplicação dinâmica. Por exemplo, o código pode ser inserido no compilador de eventos de redimensionamento de forma a reposicionar um controle para que ele permaneça centrado sobre a sua forma, expandindo-se para preencher o Form, etc. Através da inserção de código para o compilador de eventos ou uma tecla numa caixa de texto, o programa pode traduzir automaticamente, como o caso do texto que está a ser introduzido, ou mesmo impedir certos caracteres de serem inseridos.

Visual Basic pode criar ficheiros executáveis (arquivos EXE), controles ActiveX ou arquivos DLL, mas é usado principalmente para desenvolver aplicações do Windows e sistemas de interface de base de dados. As caixas de diálogo com menos funcionalidades podem ser usadas para fornecer recursos de pop-up. Controlos que fornecem a funcionalidade básica da aplicação, enquanto que os programadores podem inserir lógica adicional dentro dos manipuladores de eventos apropriados.

Por exemplo, uma caixa de combinação drop-down exhibe automaticamente uma lista. Quando o utilizador seleciona um elemento, um compilador de eventos é chamado para executar o código que o programador criou para executar a ação para o item da lista. Alternativamente, um componente do Visual Basic pode não ter nenhuma interface de utilizador, e sim fornecer objetos ActiveX para outros programas através do Component Object Model (COM), isso permite que o processamento seja do lado do servidor ou num módulo add-in.

O tempo de execução recupera a memória não utilizada usando uma contagem de referência, que depende de variáveis que passam para fora do scope ao ser definida como vazia, evitando o problema de esgotamento da memória comum em relação a outras linguagens. Há uma grande biblioteca de objetos de utilidade, e a linguagem fornece suporte básico orientado a objetos. Ao contrário de muitas outras linguagens de programação, o Visual Basic geralmente não diferencia maiúsculas de minúsculas, embora ele transforme palavras-chave numa configuração padrão de casos e força o caso de nomes de variáveis para se conformar, como o caso da entrada na tabela de símbolos. As comparações de cadeia são sensíveis por padrão.

O compilador de Visual Basic é compartilhado com outras linguagens do Visual Studio (C, C++). No entanto, por padrão, as restrições no IDE não permitem a criação de alguns alvos (Windows modelo DLLs) e de modelos de segmentação, mas ao longo dos anos, os programadores têm ignorado estas restrições.

As características do Visual Basic que diferem das línguas derivadas do C são:

- Demonstrações tendem a terminar com palavras-chave, tais como "End If", em vez de usar "{}" às declarações do grupo.
- A atribuição de variável não é possível. Se  $A = B = C$  não implica que os valores de A, B e C são iguais. O resultado booleano de "Is B = C?" é armazenado em A. O resultado armazenado em A, por conseguinte, é verdadeiro ou falso.
- Constantes booleanas verdadeiras têm valor numérico -1. Isso ocorre porque o tipo de dados booleano é armazenado como um inteiro de 16 bits. Nesta construção -1 avalia os 16 binários (o valor booleano True), e 0 até 16 0s (o valor booleano falso). Isto é evidente ao executar uma operação não em 16 bit assinando um valor inteiro 0, que retorna o valor inteiro -1, em outras palavras True = Não False. Esta funcionalidade inerente torna-se especialmente útil ao realizar operações lógicas sobre os bits individuais de um número inteiro, como E, OR, XOR e NOT. [16] Esta definição de True também é consistente com BASIC desde o início dos anos 1970.
- Os operadores lógicos bit a bit são unificados. Isso é diferente de algumas linguagens derivadas de C (tal como o Perl), que têm operadores lógicos bit a bit e separados. Esta é uma característica nova do BASIC.
- Bases de matriz variável. Arrays são declarados especificando os limites superiores e inferiores de uma forma similar ao Pascal e Fortran. Também é possível usar a instrução Option Base para definir o padrão limite inferior. O uso da instrução Option Base pode levar a confusões durante a leitura de código do Visual Basic e é melhor evitar usar explicitamente o limite inferior da matriz. Este limite inferior não é limitado a 0 ou 1, porque ele também pode ser definido por declaração. Desta forma, tanto os limites inferiores e superiores são programáveis. Em mais línguas, o limite inferior da matriz não é variável. Esta característica incomum existe no Visual Basic .NET, mas não em VBScript.
- OPÇÃO BASE foi introduzida pelo ANSI, com o padrão de ANSI para Minimal BASIC no final de 1970.
- Relativamente forte é a integração com o sistema operacional Windows e o Component Object Model. Os tipos nativos de strings e arrays são os tipos COM dedicados, BSTR e SAFEARRAY.
- Inteiros são automaticamente promovidos para reais em expressões que envolvem o operador normal de divisão (/), de modo a que a divisão de um número inteiro por outro produza o resultado intuitivamente corretas. VB fornece um operador de divisão de inteiros específicos (\).

- Por padrão, se uma variável não for declarada ou se nenhum caráter do tipo de declaração seja especificado, a variável é do tipo Variant. No entanto, este pode ser alterado com instruções DefType como DefInt, DefBool, DefVar, DefObj, DefStr. Há 12 instruções DefType no total oferecidas pelo Visual Basic 6.0. O tipo de padrão pode ser substituído por uma declaração específica, utilizando um carácter especial num sufixo no nome da variável. VB também pode ser definida num modo que só explicitamente declarando variáveis possam ser usadas com o comando Option Explicit.

## 6 Arquitetura do Sistema

Ao longo deste capítulo vão ser descritas as tecnologias envolvidas no desenvolvimento da aplicação, apresentada a interface gráfica, referidos aspetos de usabilidade, segurança, referido ainda o modelo mental esperado pelos utilizadores da aplicação, ilustrado o esboço inicial e identificadas as partes da aplicação.

A aplicação possui uma arquitetura Cliente-Servidor Web, sendo constituída por um servidor IIS que recebe pedidos do cliente e disponibiliza em tempo real as páginas Web criadas. Na Figura 10 podemos ver as componentes base da arquitetura, o motor Workflow onde são executados os fluxos de processo, o servidor Web onde está a FW Base que cria os verticais e a base de dados Caresflag partilhada pelos dois serviços. Na base de dados Caresflag estão armazenadas todas as configurações dos verticais e dos Workflows assim como os dados de negócio.

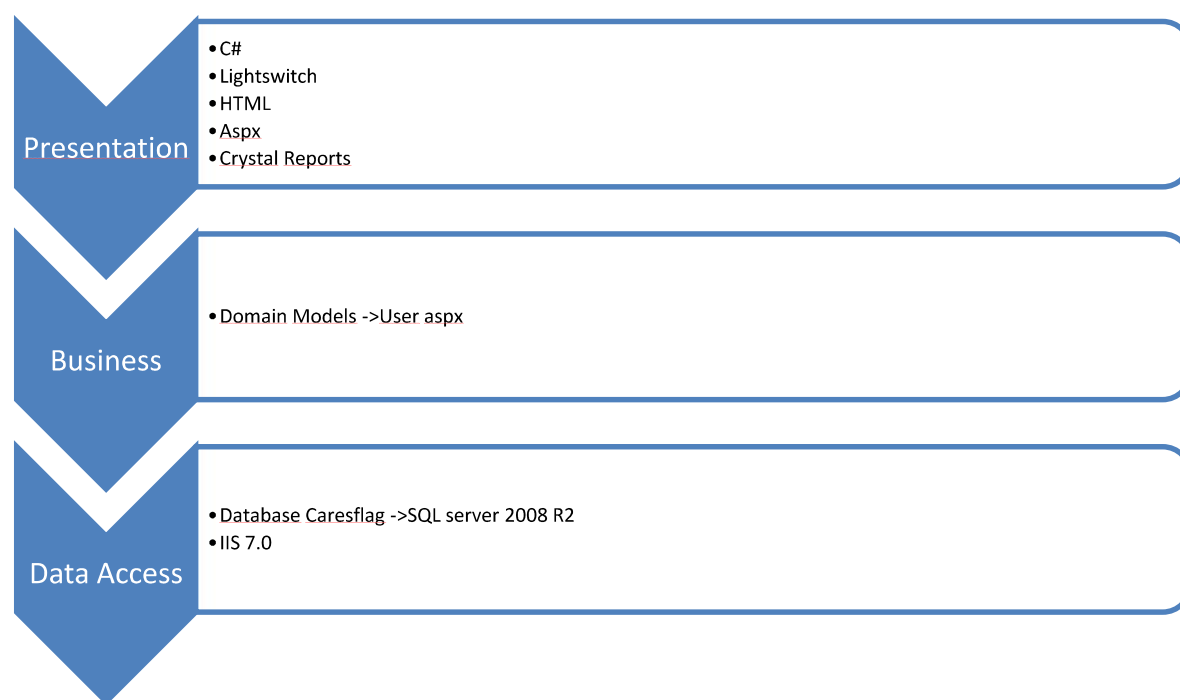


Figura 10 - Arquitetura do Software proposto na Tese

## 6.1 Modelos de Entidades e relacionamentos

Neste modelo, em primeiro lugar definem-se as entidades presentes e as relações entre elas. Este processo ajuda a perceber quais as informações que interessa guardar na base de dados.

### 6.1.1 UML

A linguagem Unified Modeling Language (UML) surge como o sucessor natural de um conjunto de métodos de análise e projecto orientados a objetos. A UML, padronizada pelo Object Management Group (OMG), é um modelo de linguagem que permite modelar os requisitos especificados. Devido ao seu elevado poder expressivo, riqueza da sintaxe e semântica, e para além da sua ampla divulgação na área de modelação de software, é propícia ao desenvolvimento de aplicações que utilizem linguagens orientadas a objetos. Os modelos apresentados são obtidos por refinamento do espaço do problema e análise das entidades, e respetivas inter-relações existentes no universo do problema a modelar. Cada modelo é constituído por um conjunto de diagramas, correspondentes a perspectivas ou pontos de vista específicos do modelo. Estes diagramas constituem representações parciais do sistema, cuja coerência semântica com os outros diagramas do mesmo modelo tem de ser assegurada. Os diagramas utilizados para a modelação deste projecto foram: diagramas caso de uso e Diagramas de sequência.

## 6.1.2 Entidades

As Entidades identificadas são descritas a seguir:

- **Utilizador:** Médico, enfermeiro, auxiliar, psicólogo, sociólogo, administração, contabilista e administrador de sistema. À exceção do administrador de sistema todos os outros são utilizadores que operam no interior da instituição, esta categoria tem como objetivo vir a ser usada pela administrador de toda a infraestrutura informática, possui acesso a toda e qualquer informação presente no sistema, tem também como função a gestão dos utilizadores que podem aceder e as respetivas permissões de acesso às diversas áreas da aplicação. (em anexo o diagrama de modelo relacional);
- **Utente:** O elemento central de toda a aplicação, este elemento está presente em praticamente todas as tabelas uma vez que só assim é possível assegurar que cada atributo pertence a qual utente. (em anexo o diagrama de modelo relacional);
- **Instituição:** A instituição tem como função a atribuição a cada utente, uma vez que se trata de uma solução integrada é necessário este elemento de forma a poder atribuir ao utente o local onde se encontra e que a qualquer momento sem perder o historial. (em anexo o diagrama de modelo relacional);
- **Equipamento:** O equipamento diz respeito aos diversos locais que uma instituição pode ter, por exemplo edifício sede e as filiais.
- **Fornecedores:** Tal como o nome indica trata-se dos diversos fornecedores que distribuem material para a instituição, a instituição também pode ser fornecedora de serviços podendo constar da tabela para prestar serviços;
- **Cientes:** No caso da própria instituição ser fornecedora de serviços fora do âmbito normal é esta entidade que se caracteriza por ser cliente dos serviços prestados pela instituição;
- **Família:** Trata-se de todos os familiares que estão ligados ao utente, de forma a poder ser relacionado com os graus de parentesco inclusive dentro da própria instituição;

### 6.1.3 Casos de uso

No sistema todas as ações requerem que o utilizador esteja autenticado. Os diagramas das figuras seguintes mostram as ações que os atores podem realizar.

Cada caso de uso é uma ação no sistema, é descrito cada um, referindo-se as ações possíveis:

- **Autenticar:** A autenticação no sistema é a pré-condição para poder aceder-lhe.
- **Consultar utilizadores:** Obter uma lista dos utilizadores registados na plataforma.
- **Gerir utilizadores:** Os administradores têm a opção de criar utilizadores e modificar os dados deles.
- **Gerir Utente:** Sobre um utente as ações possíveis são consultar, criar, modificar e anular.
- **Consultar Utente:** Incluem dados que são geridos e dados exclusivamente informativos em que a sua gestão inclui apenas consultá-los.
- **Efetuar registos:** Colocar ou retirar registos de análise aos utentes, por exemplo medição de tensão arterial.

### 6.1.3.1 Administrador de Sistema

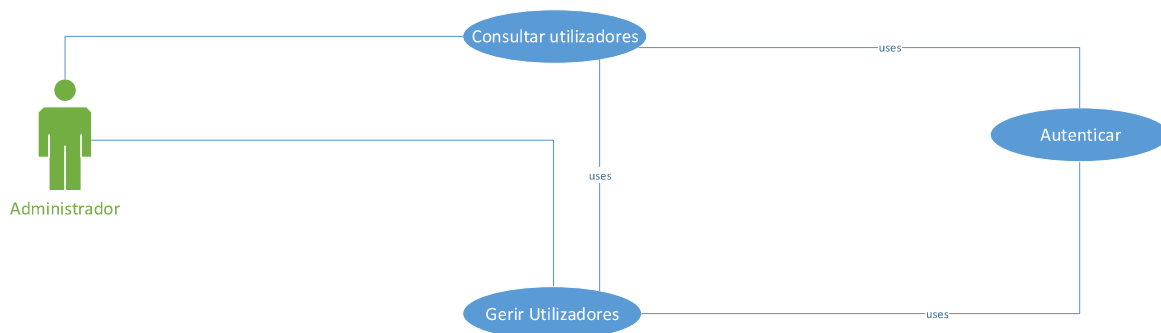


Figura 11 - Caso de uso Administrador

### 6.1.3.2 Médico

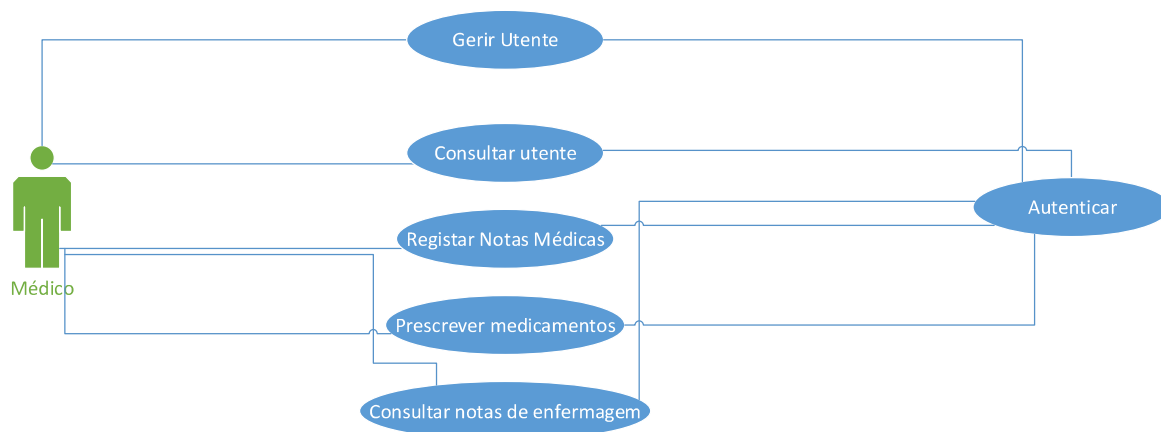


Figura 12 - Caso de uso Médico

### 6.1.3.3 Enfermeiro



Figura 13 - Caso de uso Enfermeiro

### 6.1.3.4 Psicólogo

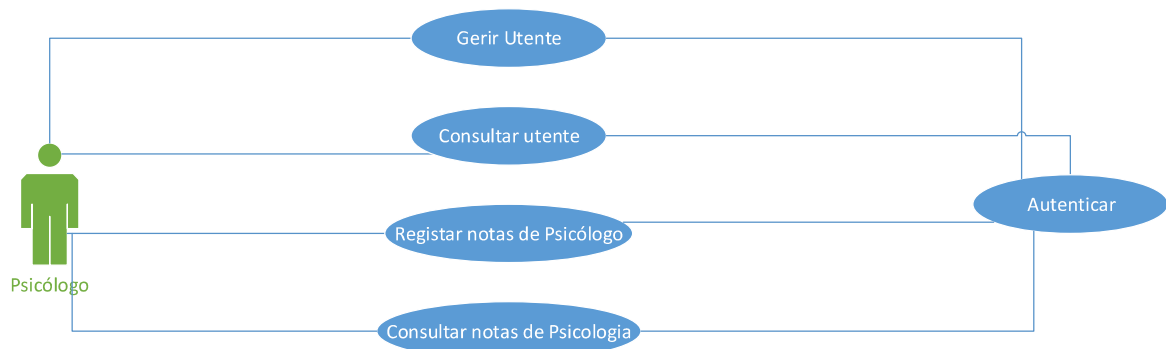


Figura 14 - Caso de uso Psicólogo

### 6.1.3.5 Sociólogo



Figura 15 - Caso de uso Sociólogo

### 6.1.3.6 Receção

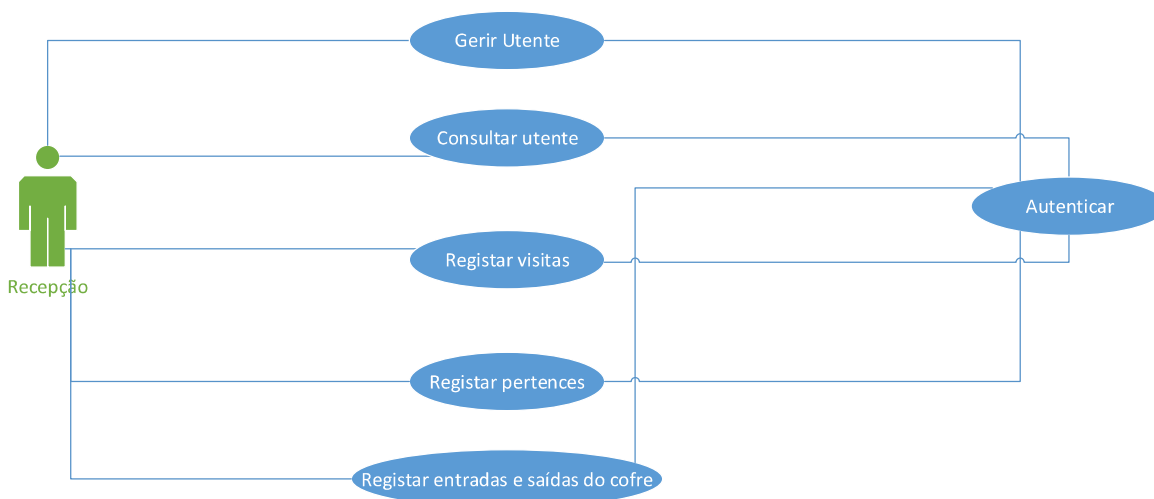


Figura 16 - Caso de uso Receção

### 6.1.3.7 Farmácia



Figura 17 - Caso de uso Farmácia

### 6.1.3.8 Apoio Social



Figura 18 - Caso de uso Apoio Social

#### 6.1.4 Diagramas de Sequência

Diagrama de sequência é um diagrama usado em UML, representando a sequência de processos (mais especificamente, de mensagens passadas entre objetos) num programa de computador. Como um projeto pode ter uma grande quantidade de métodos em classes diferentes, pode ser difícil determinar a sequência global do comportamento. O diagrama de sequência representa essa informação de uma forma simples e lógica.

Um diagrama de sequência descreve a maneira como os grupos de objetos colaboram em algum comportamento ao longo do tempo. Ele registra o comportamento de um único caso de uso e exibe os objetos e as mensagens passadas entre esses objetos no caso de uso.

Em síntese: o Diagrama de Sequência é uma das ferramentas UML usadas para representar interações entre objetos de um cenário, realizadas através de operações ou métodos (procedimentos ou funções). Este diagrama é construído a partir do Diagrama de Casos de Usos.

Primeiro, define-se qual o papel do sistema (Use Cases), depois, é definido como o software realizará seu papel (Sequência de operações).

O diagrama de sequência dá ênfase á ordenação temporal em que as mensagens são trocadas entre os objetos de um sistema. Entende-se por mensagens os serviços solicitados de um objeto a outro, e as respostas desenvolvidas para as solicitações.

### 6.1.4.1 Autenticar

Como mostra a Figura 19, para um utilizador se autenticar será necessário preencher um formulário com os campos de nome de utilizador e de palavra-passe com as suas credenciais.

Depois de submeter os dados introduzidos existirão duas alternativas, ou o utilizador entra no sistema, ou será apresentada uma mensagem de erro. Na segunda alternativa poderão ser apresentadas várias mensagens, quando o sistema não encontra nenhum utilizador com as credenciais introduzidas, quando encontra mas não tem permissão para entrar.

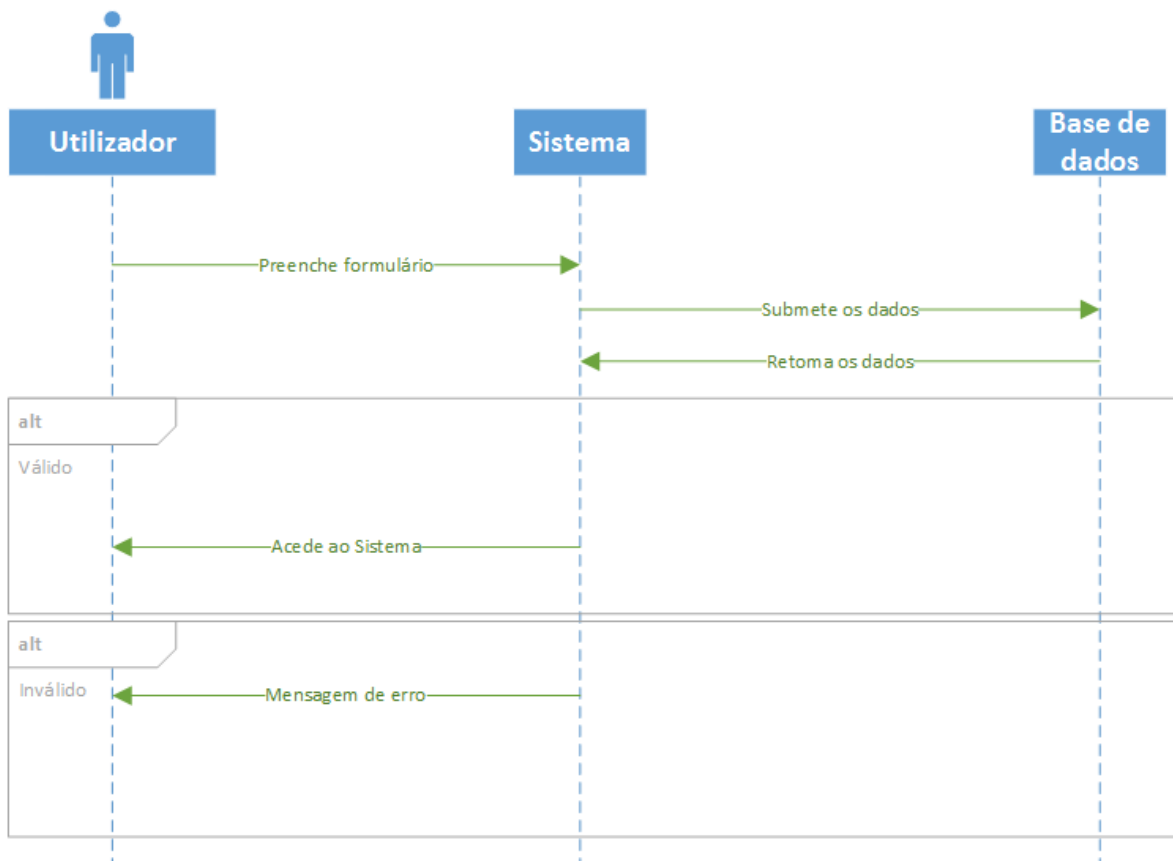


Figura 19 - Diagrama de Sequência Autenticação

### 6.1.4.2 Criar Utente

Na figura 20 podemos verificar o procedimento para criar novo utente, é uma tarefa bastante fácil de aceder uma vez que existe um botão próprio para o efeito que remete á ficha de novo utente onde o utilizador irá criar o utente com os devidos parâmetros preenchidos.

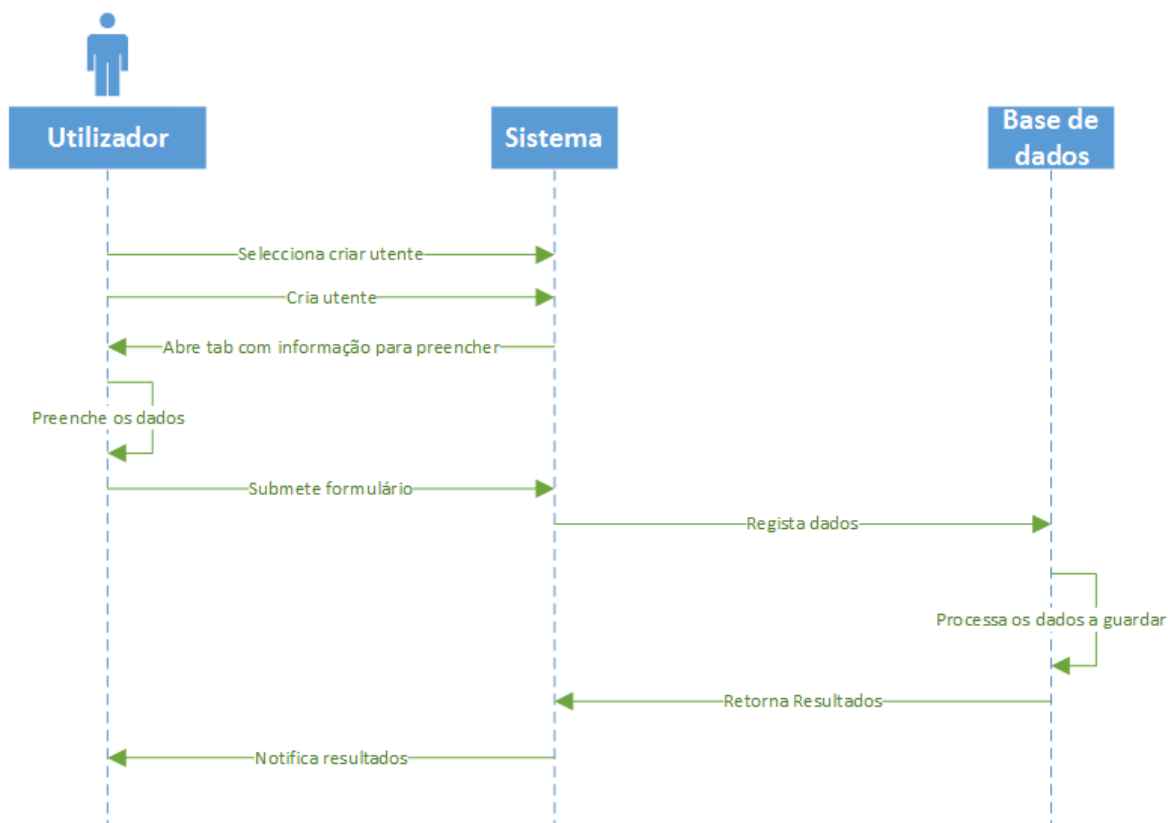


Figura 20 - Diagrama de Sequência criar utente

### 6.1.4.3 Registrar Alterações

Um utilizador que necessite de entrar para efetuar alterações no utente, de forma a adaptar a uma nova realidade ou alguma alteração que se verifique em termos de observação do mesmo. Na figura 21 podemos verificar os passos necessários para registar as alterações, um passo necessário sempre que for preciso alterar algum dado é escolher o utente que pretendemos alterar.

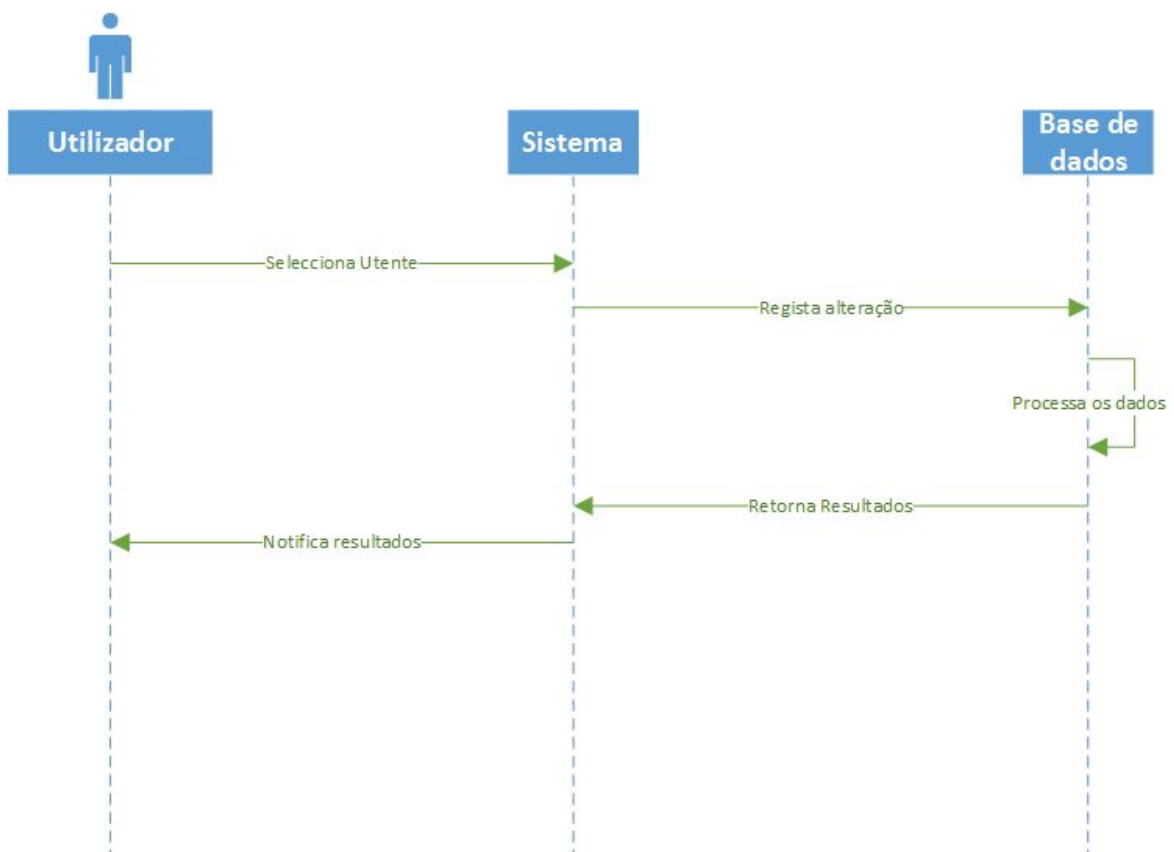


Figura 21 - Diagrama de Sequência Registrar Alteração

### 6.1.4.4 Criar Instituição

Na figura 22 podemos verificar a forma de criar uma nova instituição, esta parte tem uma importância acrescida uma vez que todos os movimentos efetuados na aplicação têm que ser relacionados com a instituição, pois ao tratar-se de um sistema integrado com outras instituições não correr o risco de adicionar utentes a outra instituição.

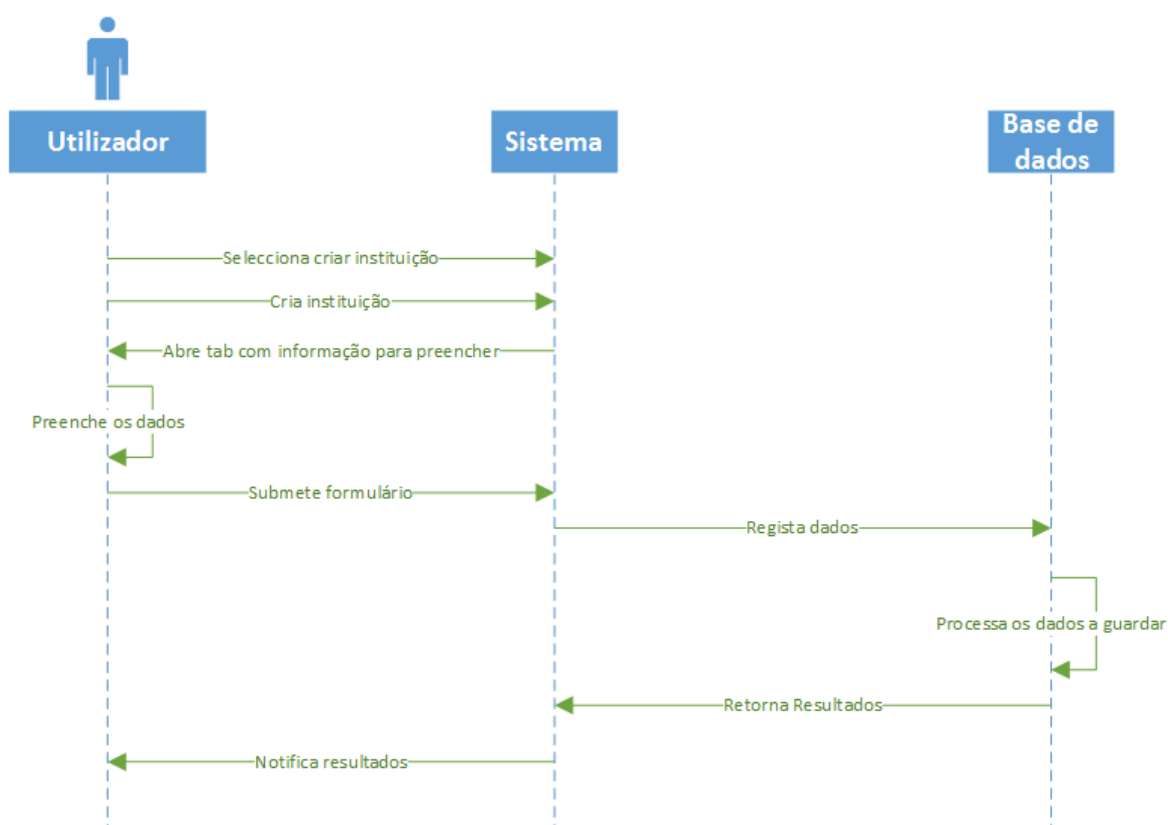


Figura 22 - Diagrama de Sequência criar instituição

### 6.1.4.5 Registos de Monitorização

A figura 23 mostra o diagrama responsável por um dos ecrãs com mais acessos da aplicação uma vez que se trata do ecrã onde é feito todos os registos de monitorização do utente.

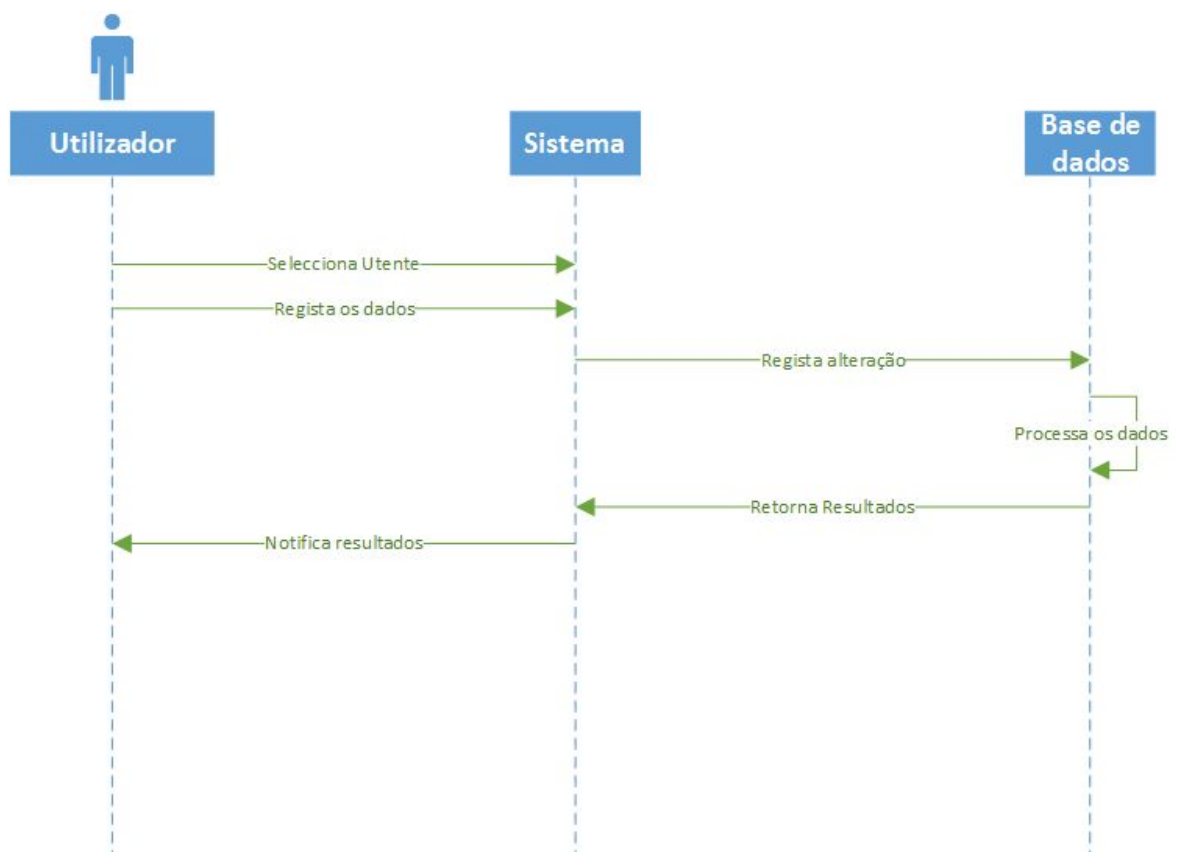


Figura 23 - Diagrama de Sequência registo de monitorização

## 7 Funcionalidades implementadas

Vão ser descritas as funcionalidades implementadas, bem como referidas as funcionalidades que não foram implementadas e que poderão ser desenvolvidas numa futura versão da aplicação. Nesta secção serão exibidas imagens da aplicação que foi desenvolvida.

A figura 24 apresenta o ecrã de login onde o utilizador irá colocar a sua informação de entrada. Trata-se de um ecrã bastante simples e intuitivo de forma a não haver problemas na utilização do mesmo.

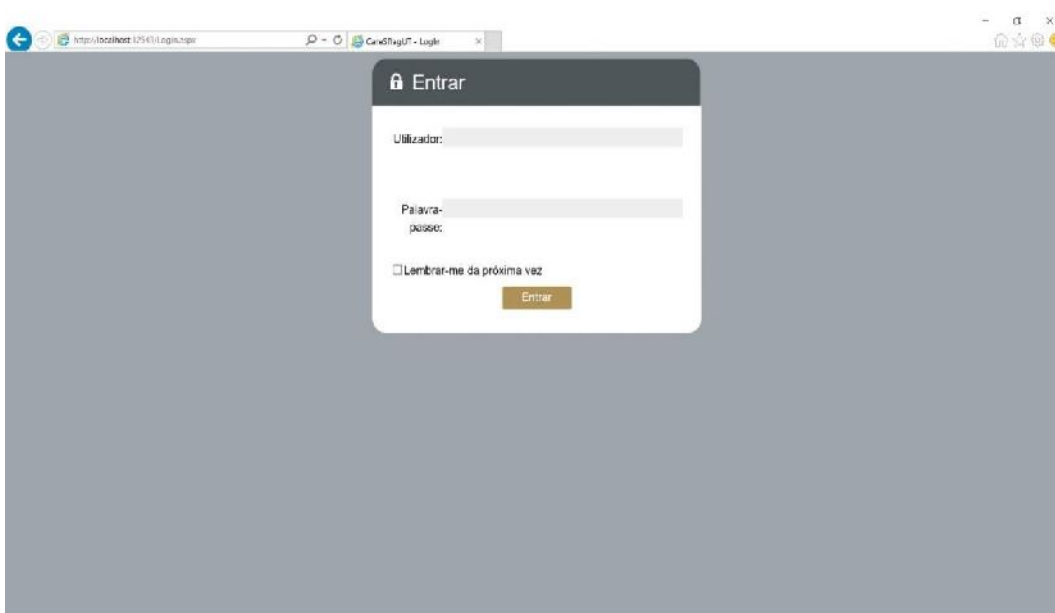


Figura 24 - Ecrã de login

A figura 25 é o ecrã principal quando o utilizador dá entrada na aplicação consegue ter-se uma visão geral sobre os utentes que fazem parte da instituição e acesso a todos os menus do lado esquerdo que permitem executar todas as tarefas desejadas pelo utilizador

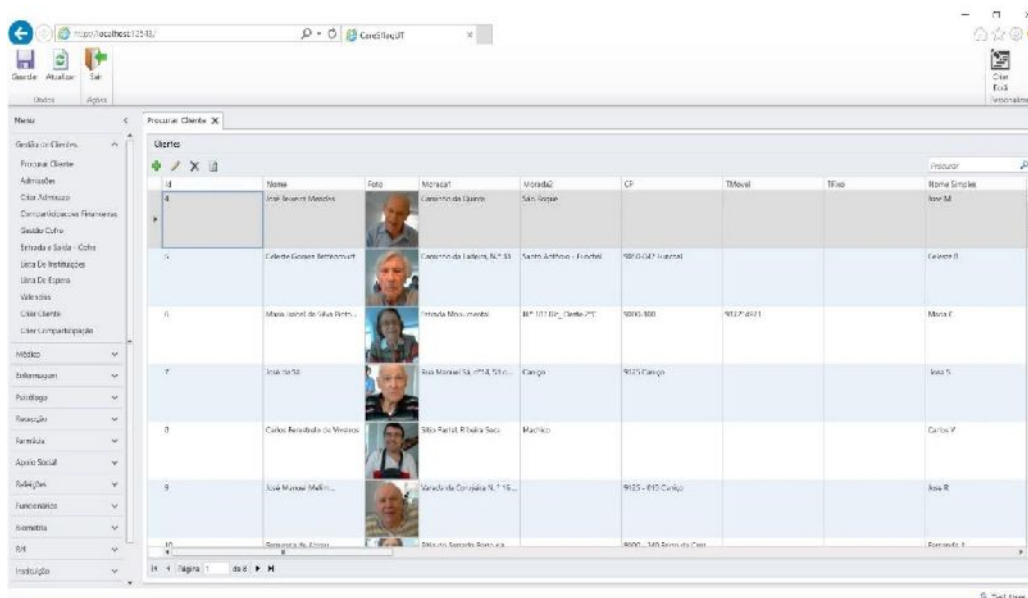


Figura 25 - Procurar Cliente

Na figura 26 é possível visualizar todos os clientes que ainda não fazem parte da instituição e que estão na fase da lista de espera, através de um simples Click é possível passar o cliente para utente da instituição e assim constar dos ecrãs internos para controle.

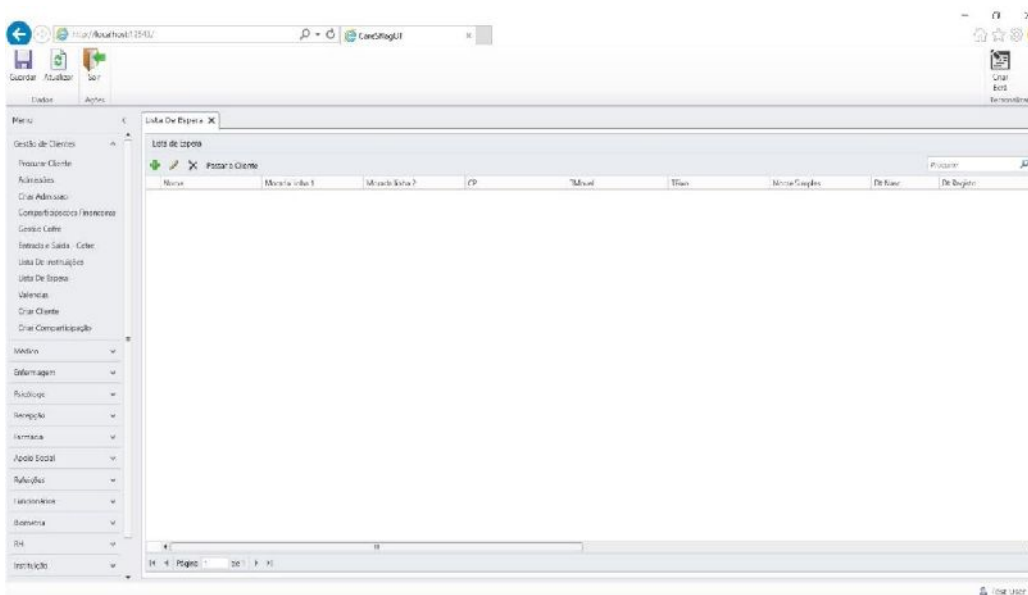


Figura 26 - Lista de espera

A figura 27 mostra o ecrã responsável pela criação de um cliente novo, quando não se encontra em nenhuma lista de espera é necessário criá-lo e é este o local próprio.

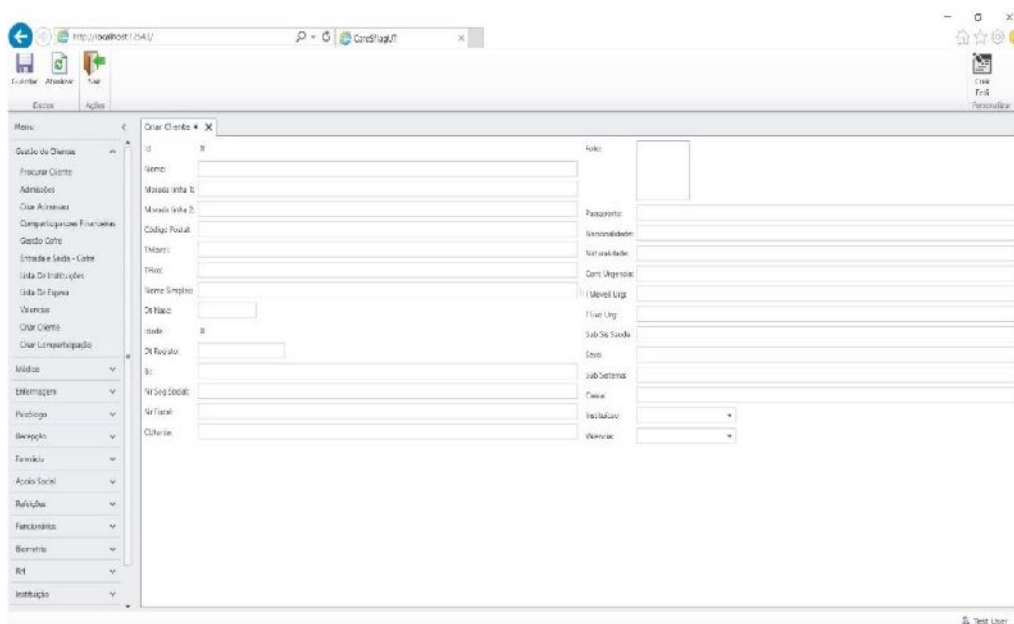


Figura 27 - Criar Cliente

Na figura 28 consegue-se atribuir a capacidade física e funcional do utente de forma a ser mais fácil de identificar o tipo de recuperação e acompanhamento a ser feito pela instituição.

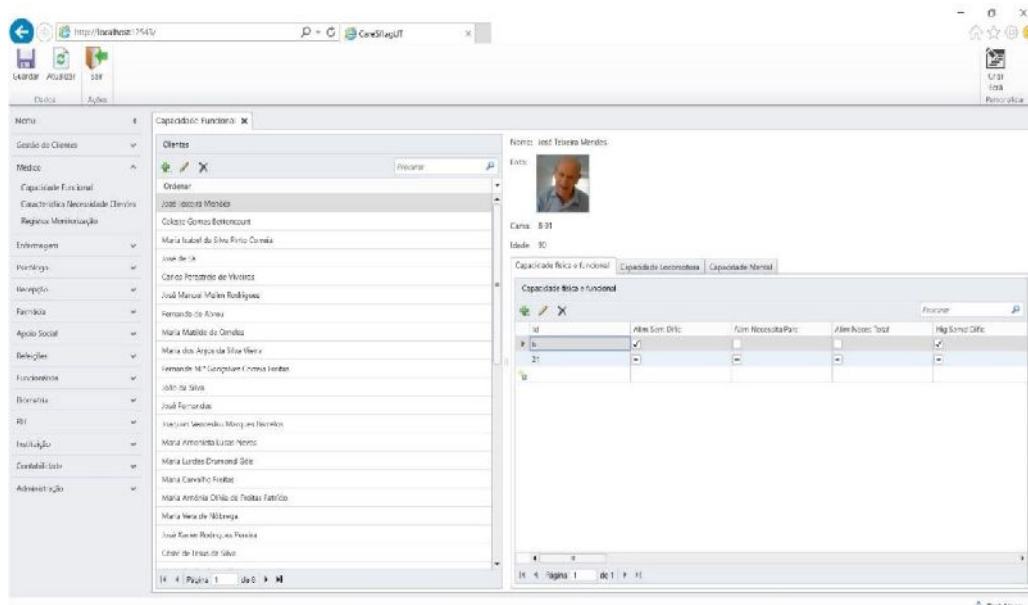


Figura 28 - Características físicas e funcionais

A figura 29 mostra a capacidade locomotora onde podemos colocar a forma física em termos de deslocação que o utente consegue ou não fazer.

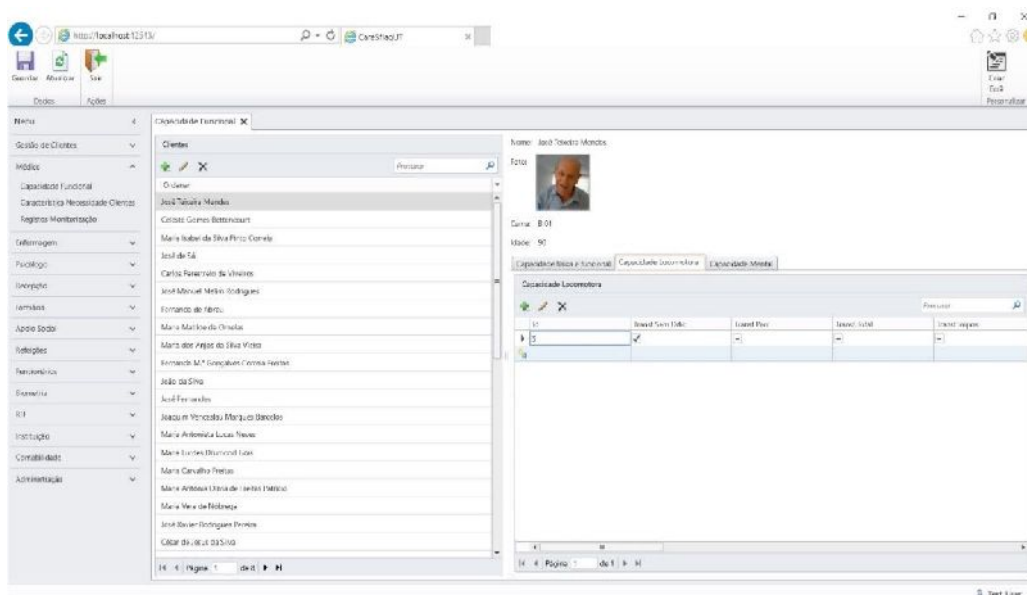


Figura 29 - Capacidade locomotora

Devido a idade avançada é necessário verificar a capacidade mental em que o idoso se encontra sendo assim é na figura 30 que mostra o ecrã onde é feita a classificação.

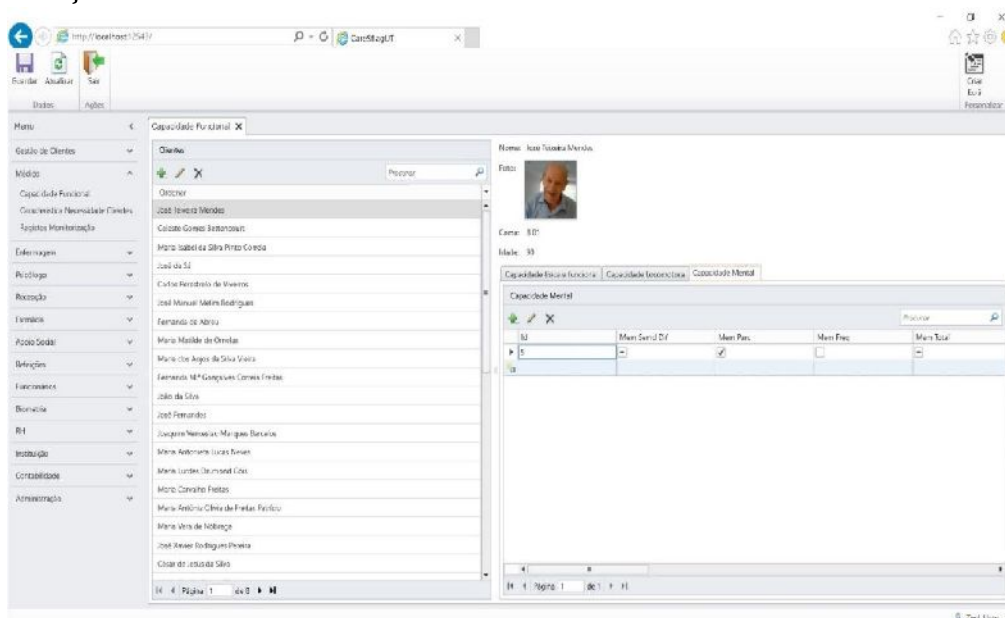


Figura 30 - Capacidade mental

Na figura 31 temos o ecrã onde é feita a avaliação geral ao utente sendo assim conseguimos classificar o utente em diversos graus de dependência.

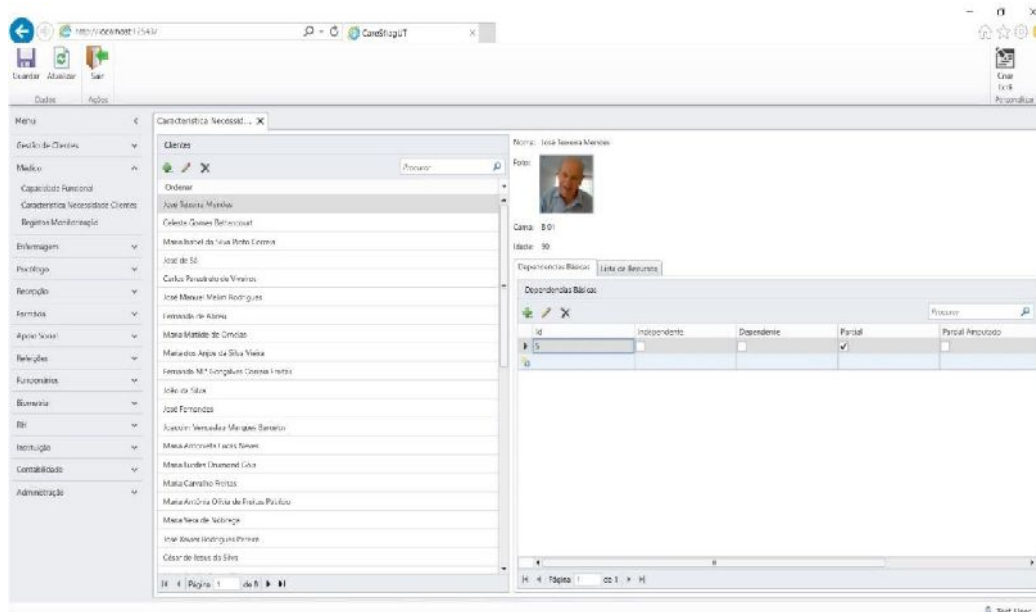


Figura 31 - Dependência física

É importante para a instituição saber o percurso do utente de onde veio para onde foi e ficar com o histórico mesmo após o utente ter saído da instituição, uma vez que pode sempre voltar a integrar a mesma, a figura 32 ilustra isso mesmo.

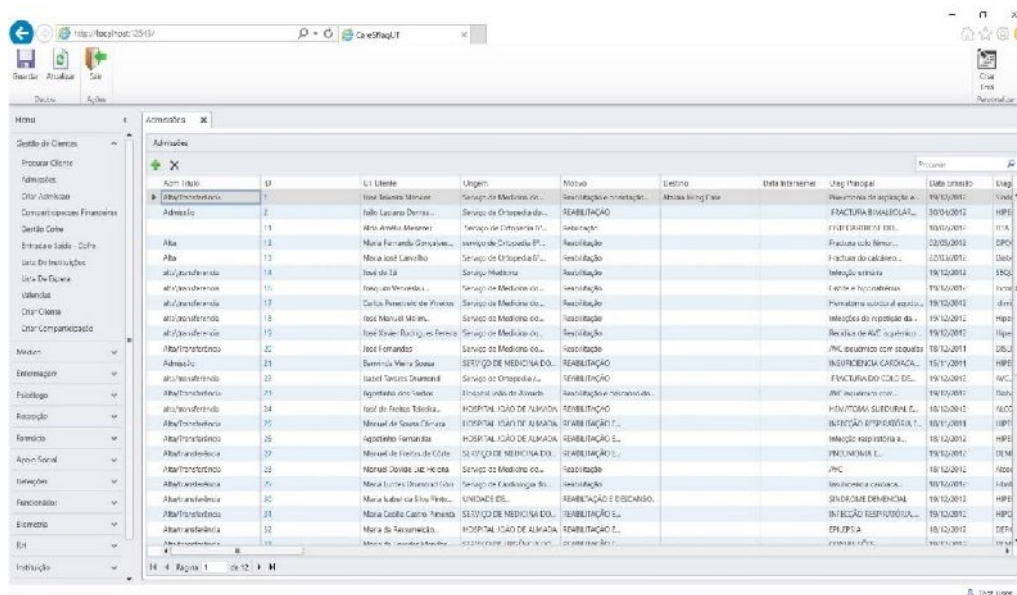


Figura 32 - Admissões

É possível no ecrã da figura 33 registar a admissão onde associamos ao utente o diagnóstico que vem no seu processo e os diferentes tipos de alta que possam fazer parte do mesmo processo.

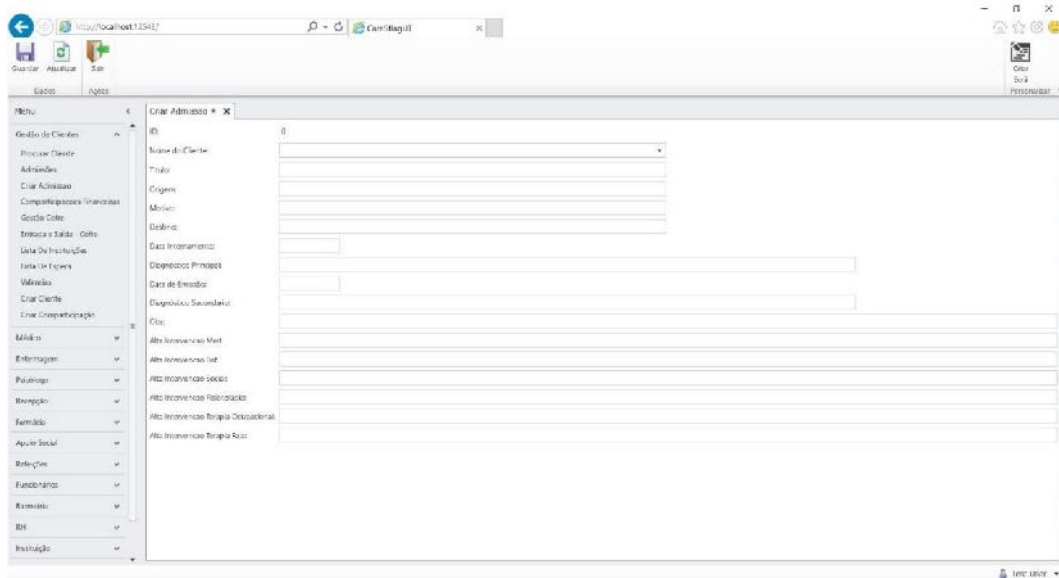


Figura 33 - Criar admissões

Já dentro da instituição é necessário registar os diferentes níveis que são atribuídos ao utente, desde tensão arterial, glicémia, etc. No ecrã da figura 34 é possível verificar os diversos registos já efetuados no utente.

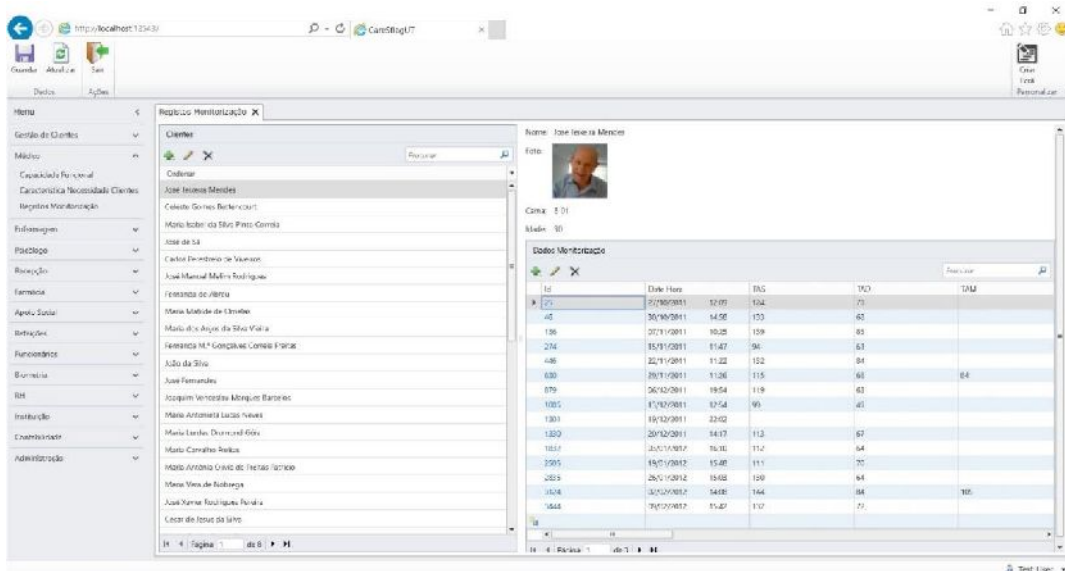


Figura 34 - Dados de monitorização



Como algumas instituições têm psicólogo é necessário haver o registo da sessão do mesmo. O ecrã da figura 37 ilustra o registo.

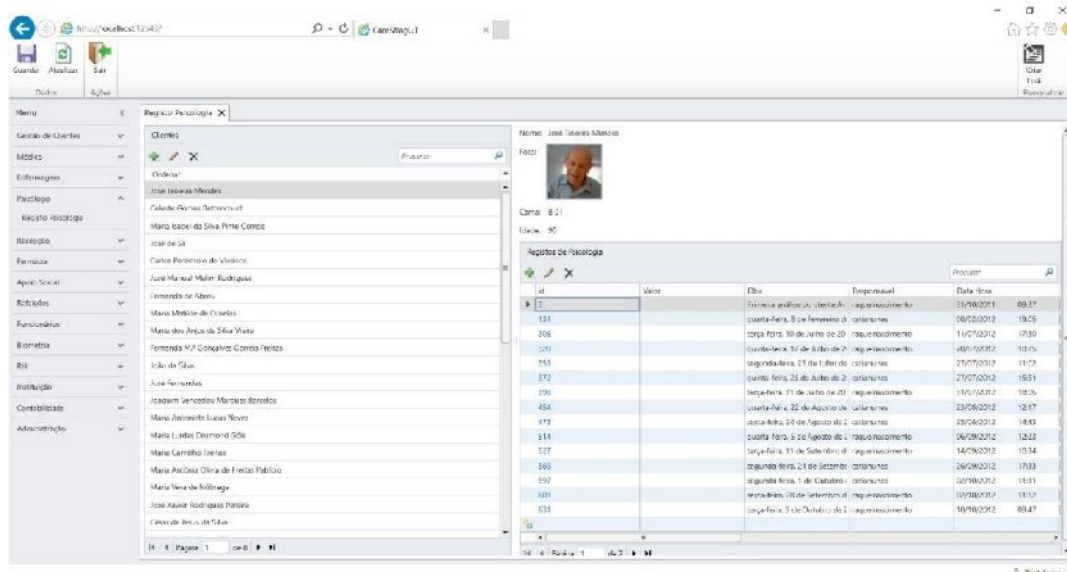


Figura 37 - Registo de psicologia

No ecrã da figura 38 é possível verificar a tabela das refeições é que pode ser editada de acordo com os dados da instituição.

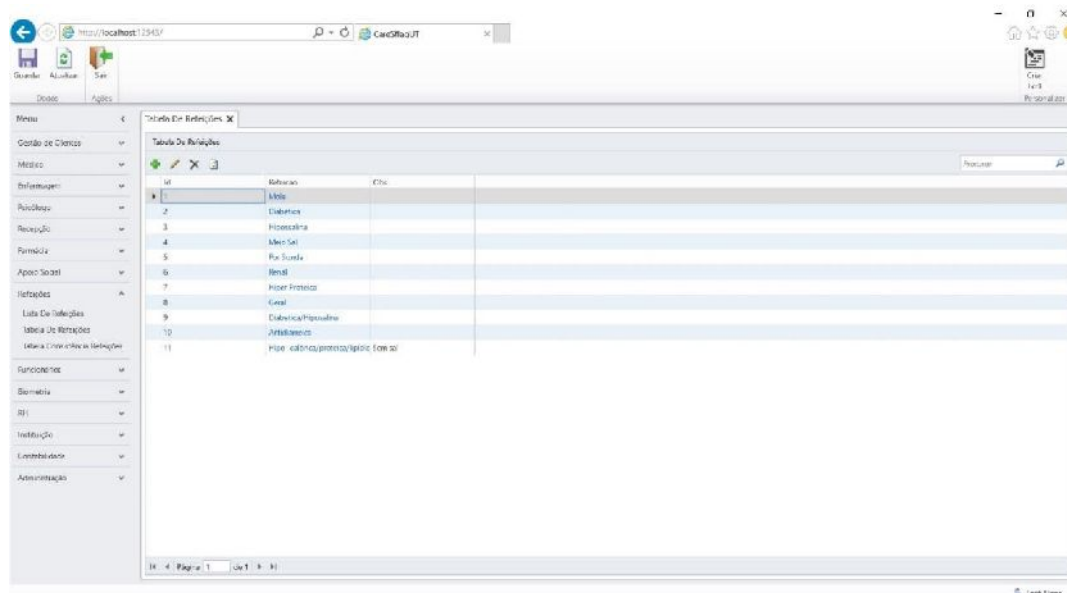


Figura 38 - Tabela de refeições

No ecrã da figura 39 mostra que conseguimos ainda definir a consistência das refeições dos utentes.

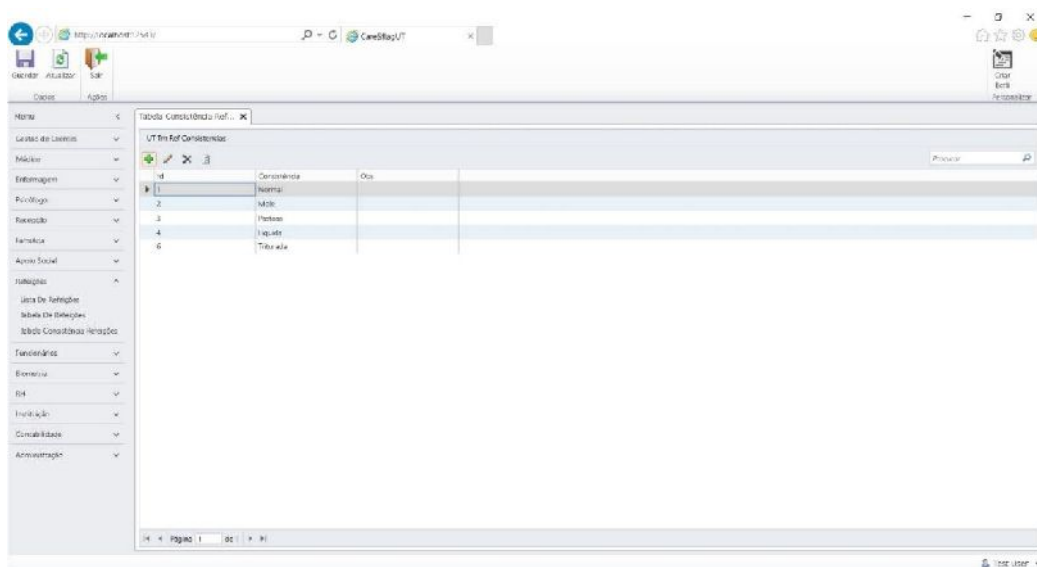


Figura 39 - Tabela de consistência

No ecrã da figura 40 é possível verificar e criar a refeição mais indicada ao utente de acordo com o decidido pelo pessoal competente

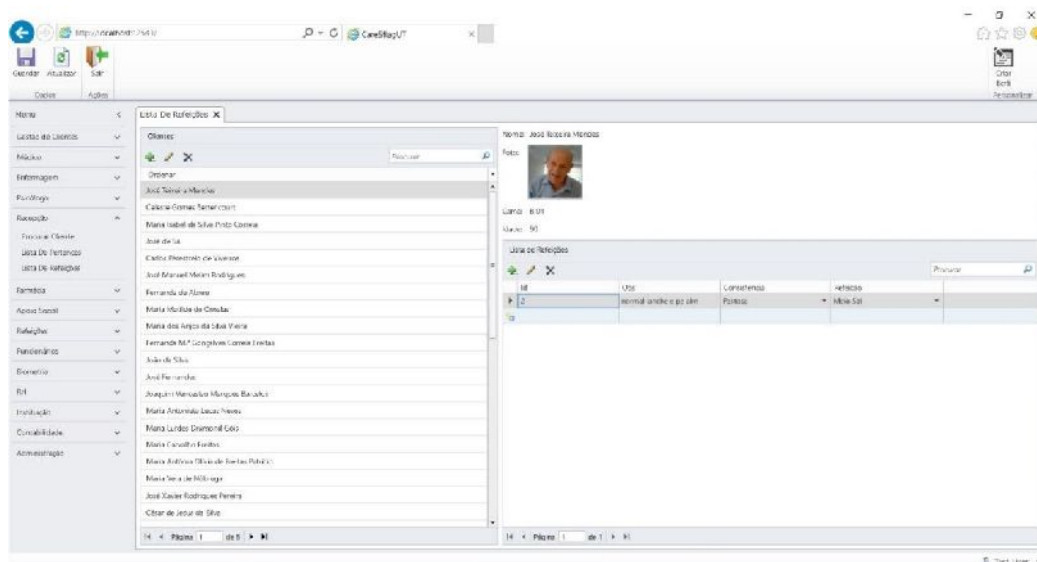


Figura 40 - Lista de refeições

É importante nos dias que correm ser feito um acompanhamento da situação social do utente de forma a poder saber a realidade que estamos a enfrentar, sendo assim

aqui é possível a assistente social colocar os registos das visitas efetuadas. No ecrã da figura 41 é possível visualizar o descrito.

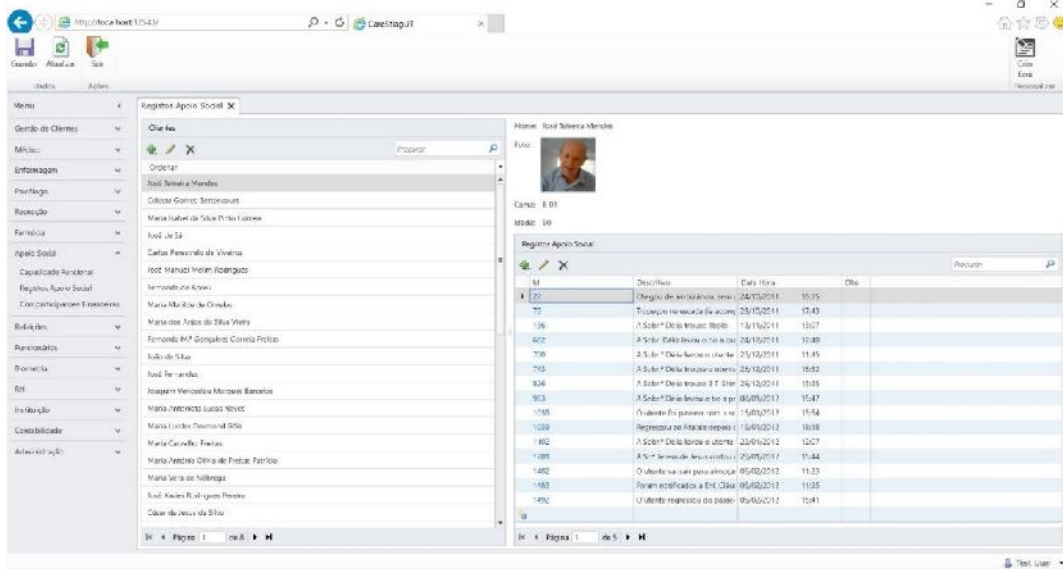


Figura 41 - Registos de apoio social

Um problema com que a instituição se depara é com a identificação dos bens dos utentes sendo assim foi criado o ecrã da figura 42 de forma a colocar o que pertence a cada um.

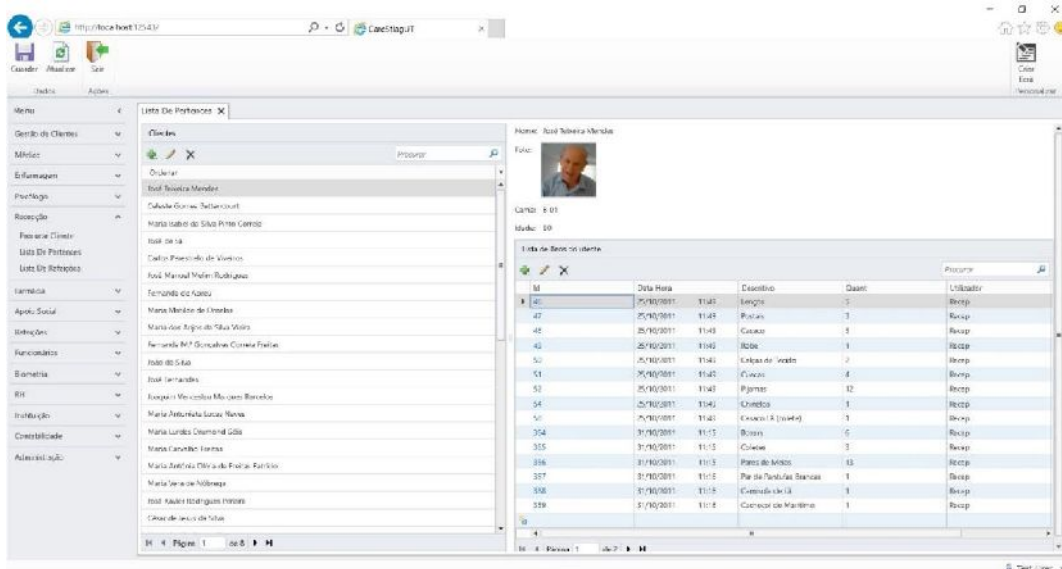


Figura 42 - lista de bens do utente

Além dos bens materiais do utente há também os bens monetários sendo assim foi criado um ecrã de gestão do cofre conforme podemos visualizar na figura 43 de forma a registar todos os movimentos.

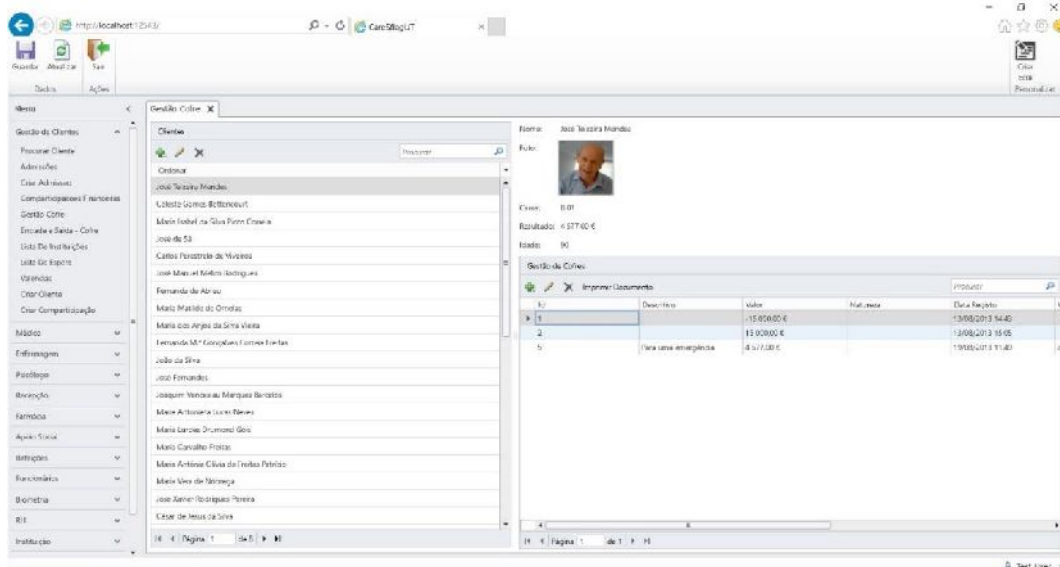


Figura 43 - gestão do cofre

Conseguimos no ecrã da figura 44 registar o movimento no cofre e para qual utente, é possível ainda identificar a barra a vermelho que diz respeito não a um problema com a aplicação mas sim com uma indicação ao utilizador sobre a forma como deve ser preenchido o ecrã a fim de prevenir erros.

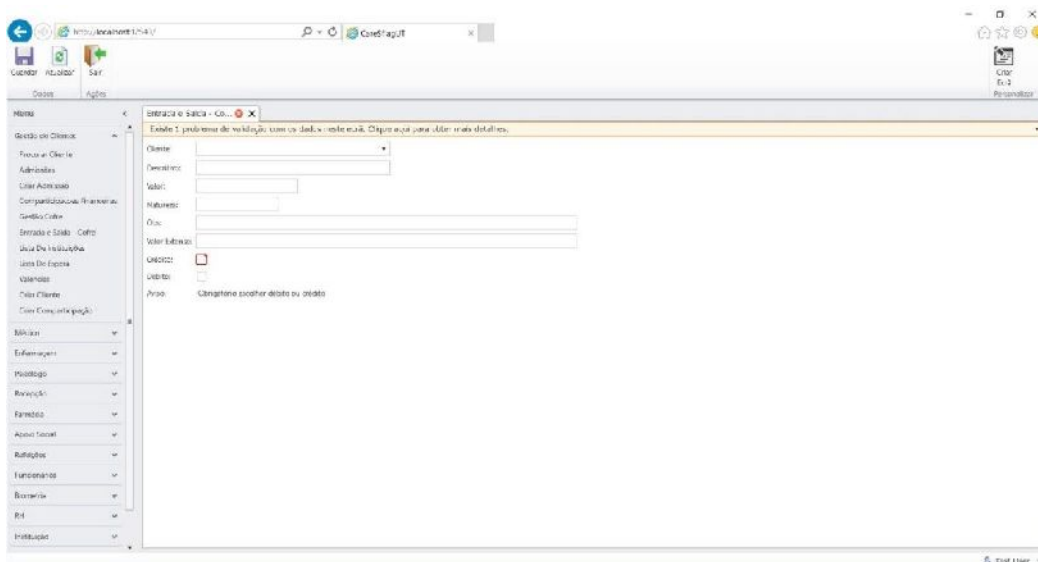


Figura 44 - entrada e saída do cofre

Podemos criar no ecrã da figura 45 as diversas valências da instituição, desde SAD, lar, centro de dia, etc...

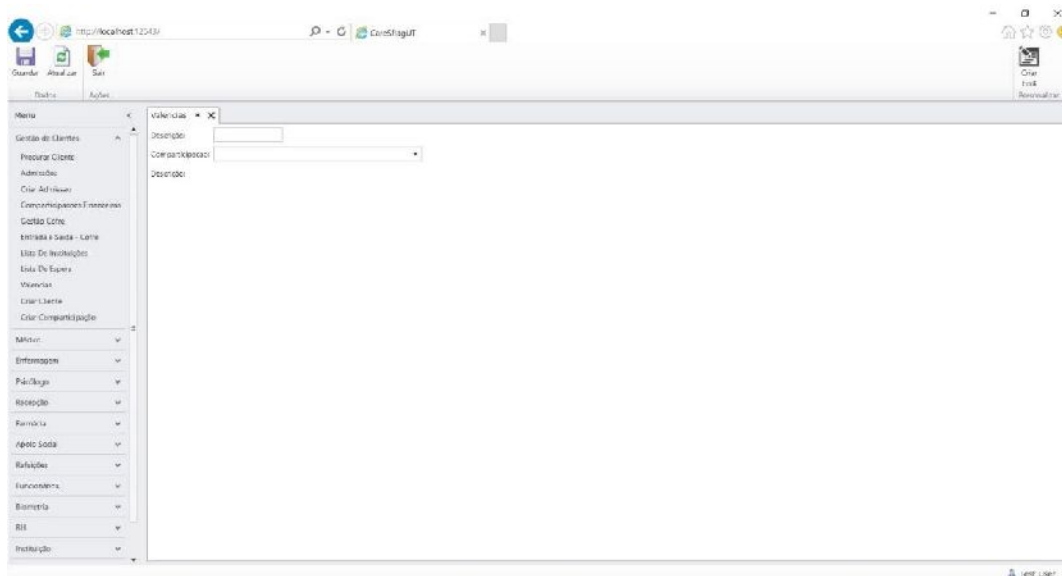


Figura 45 - criar valências

Conseguimos criar no ecrã da figura 46 as diversas instituições que compõem a que está a ser configurada em causa, ou seja, desta forma é possível criar os diversos tipos que fazem parte de uma só.

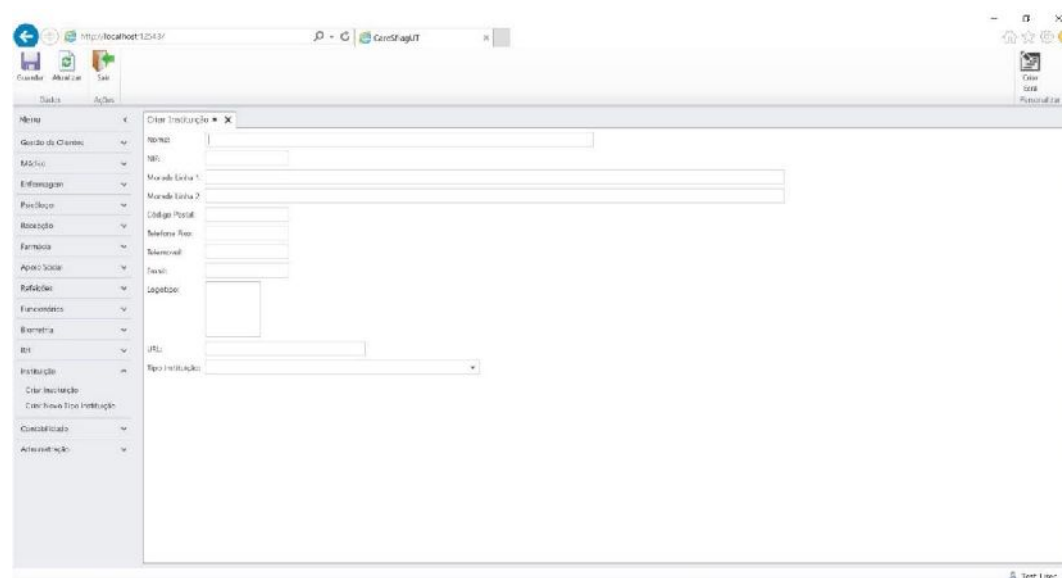


Figura 46 - criar instituições

Podemos classificar as instituições em tipos, sendo assim no ecrã da figura 47 conseguimos criar os novos tipos de instituição existentes.

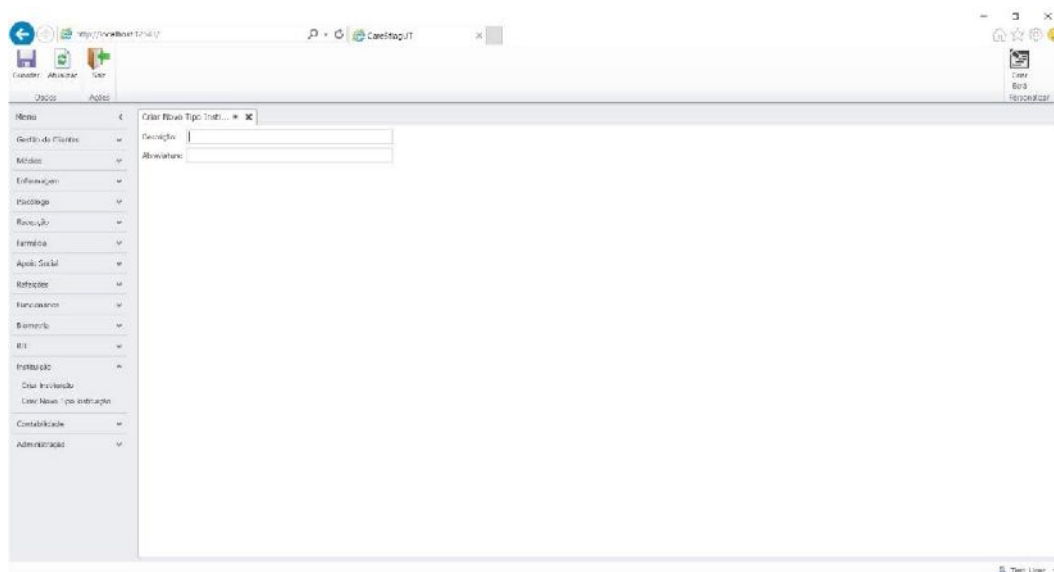


Figura 47 - criar tipo de instituições

Como algumas instituições são comparticipadas pela segurança social é importante saber o valor atribuído a cada valência sendo assim no ecrã da figura 48 podemos colocar os diversos valores e alterar sempre que for necessário.

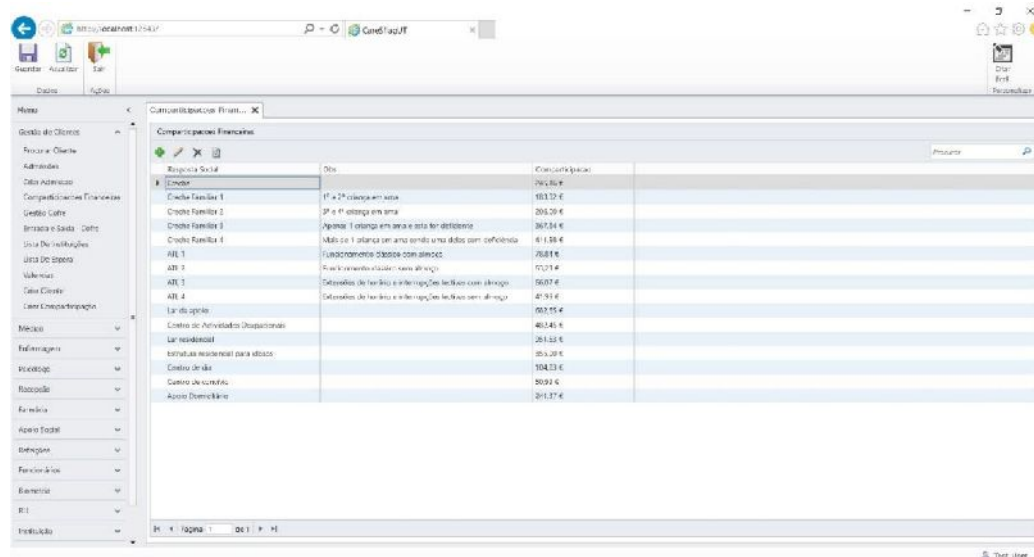


Figura 48 - comparticipações financeiras

No ecrã da figura 49 podemos criar a participação que foi referenciada no ecrã acima.

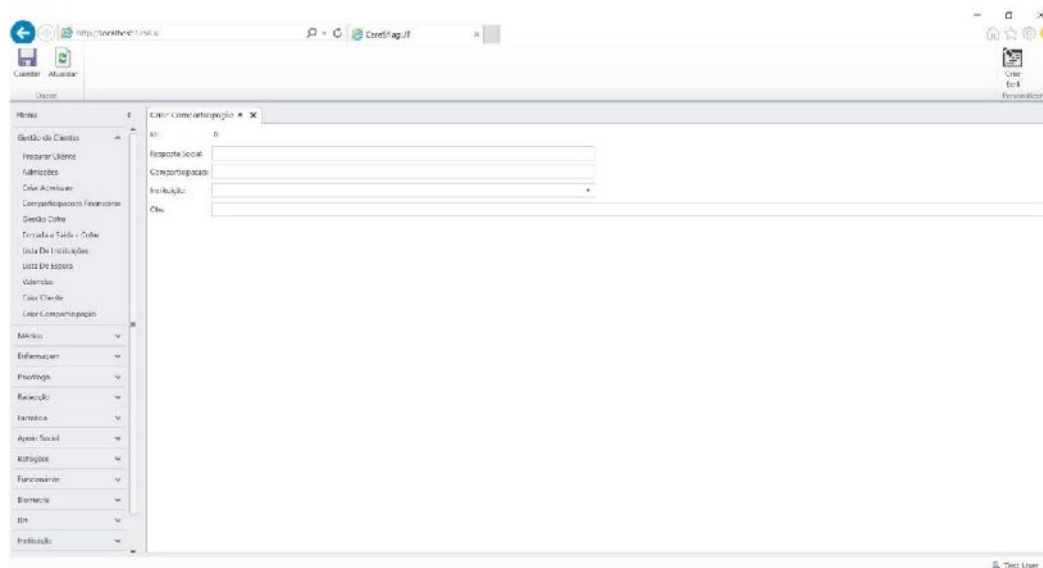


Figura 49 - criar participação

É possível no ecrã da figura 50 verificar a lista de instituições e os utentes a que cada um faz parte, é possível ainda validar o número de vagas existentes.

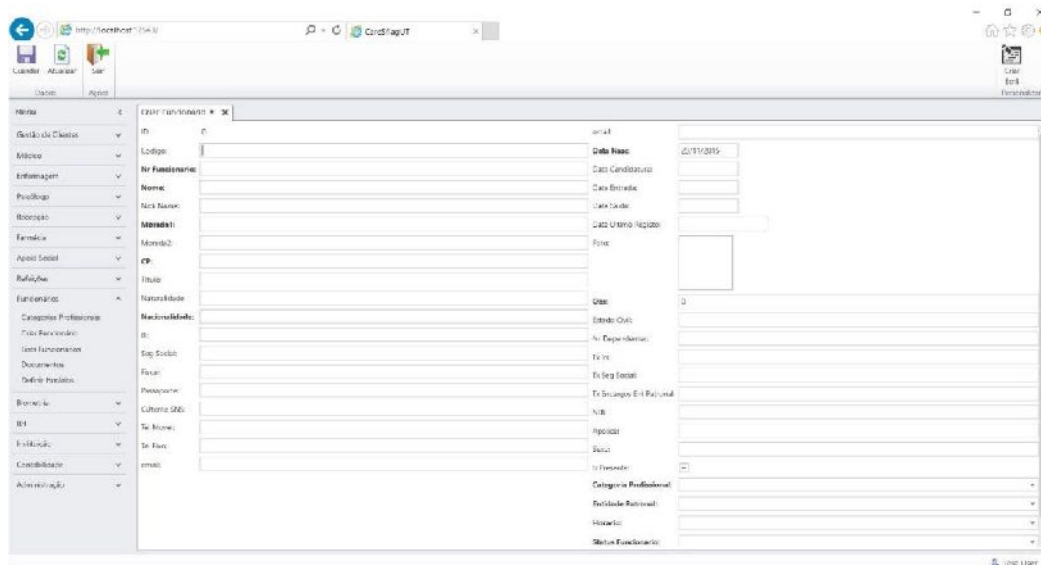


Figura 50 - criar funcionário

Tal como qualquer outra empresa esta também é composta por funcionários, sendo assim no ecrã da figura 51 podemos criar o funcionário na instituição e atribuí-lo a uma já existente.

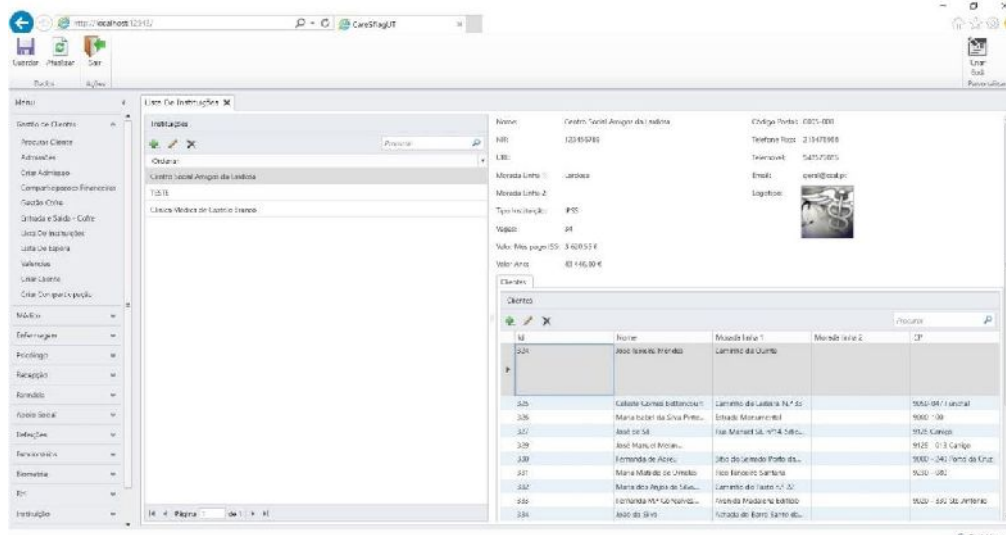


Figura 51 - clientes por instituições

No ecrã da figura 52 temos uma vista geral sobre todos os funcionários existentes na instituição.

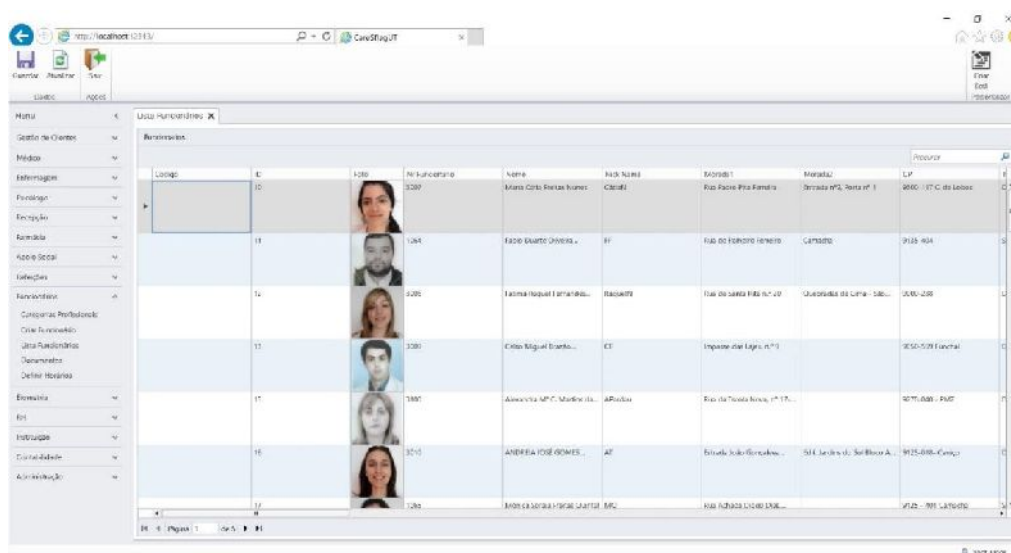


Figura 52 - lista de funcionários

No ecrã da figura 53 podemos identificar mais facilmente os funcionários tendo em conta a sua categoria profissional.

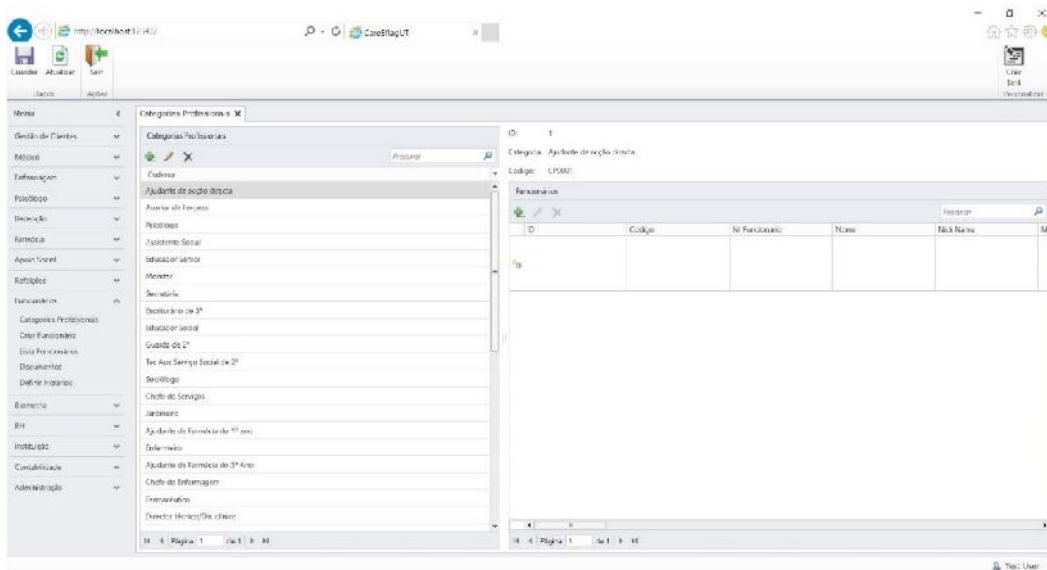


Figura 53 - funcionários por categoria

Como cada funcionário deve fazer-se acompanhar dos documentos mas que por vezes é esquecido foi criado um ecrã como podemos ver na figura 54, onde podemos efetuar o upload dos documentos pessoais e desta forma consultar sempre que for necessário.

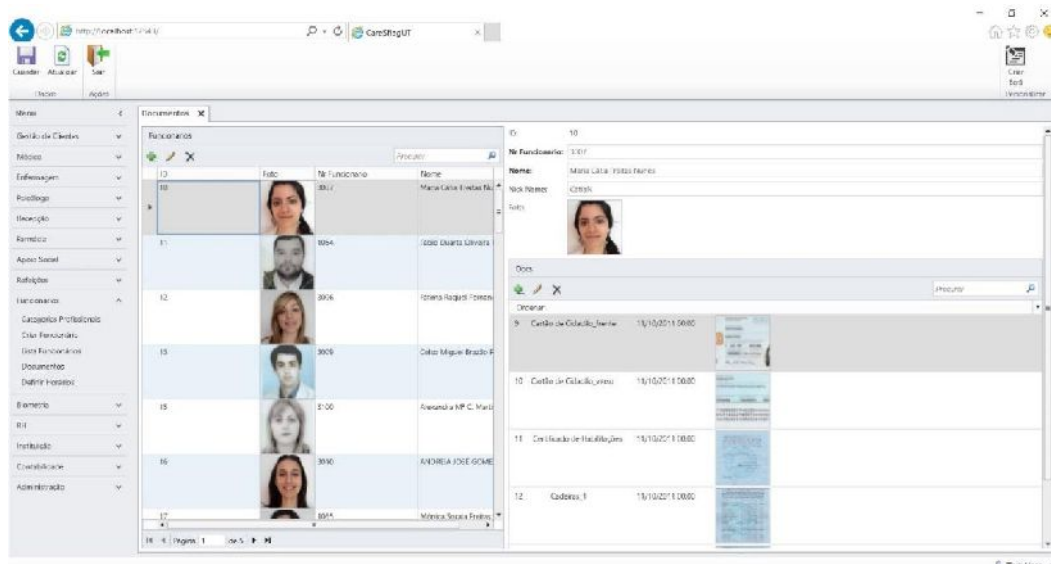


Figura 54 - documentos dos funcionários

No ecrã da figura 55 é consultado o registo biométrico da instituição de forma a saber os horários que cada um faz, esta tabela é a mesma usada em questão pelos biométricos de forma a poder ser feita a integração automática.

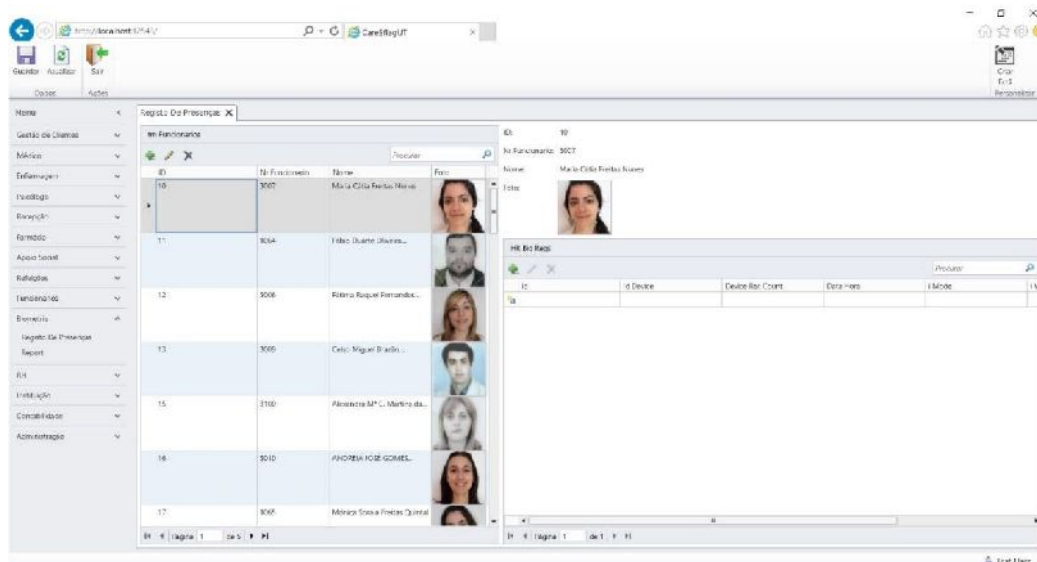


Figura 55 - registos biométricos

Através do ecrã da figura 56 podemos criar ou alterar os tipos de horários que posteriormente serão atribuídos aos funcionários.

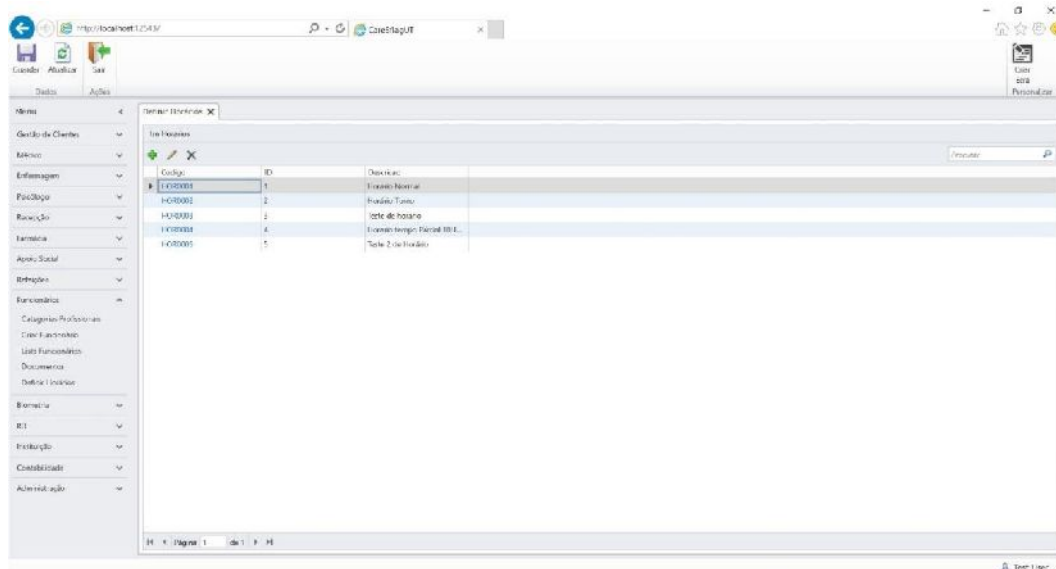


Figura 56 - definir horários

Tal como qualquer outra empresa as entradas e saídas de materiais são essenciais para o bom funcionamento da instituição, sendo assim criamos os diversos ecrãs para o controle, no caso do ecrã da figura 57 o de criar fornecedor com as características necessárias para o caso.

Figura 57 - criar fornecedor

Com o ecrã da figura 58 podemos criar as faturas necessárias, não só de produtos mas também de serviços prestados.

Figura 58 - criar fatura

No ecrã da figura 59 podemos criar produtos novos com os devidos custos associados.

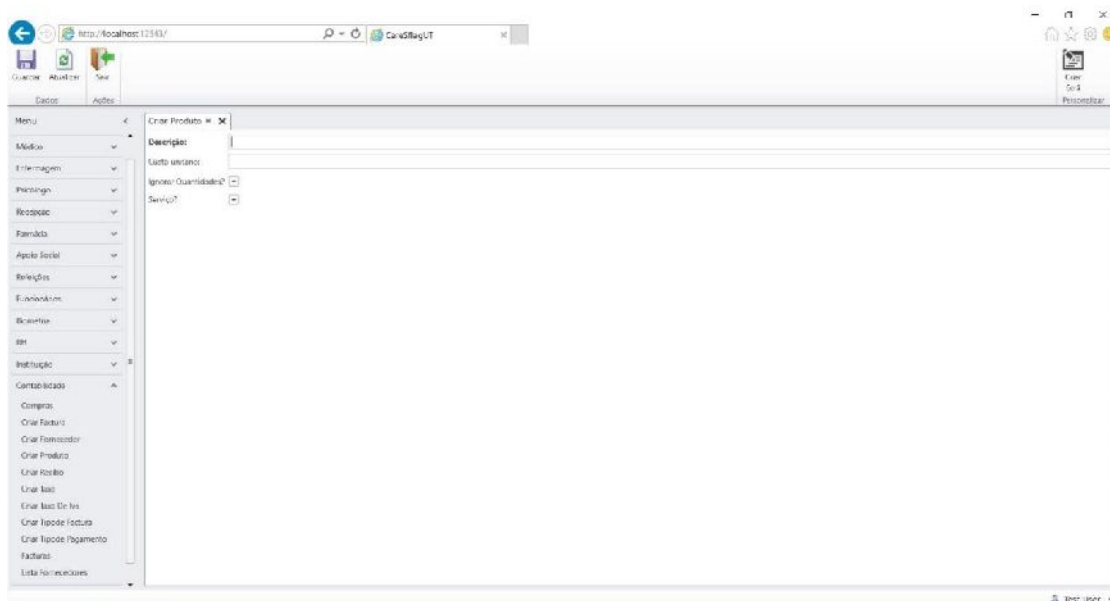


Figura 59 - criar produto

No ecrã da figura 60 é referido as faturas, como as boas práticas da contabilidade indicam deverá ter um recibo para cada, sendo assim no ecrã da figura 49 podemos escolher a fatura e processar o recibo em causa.

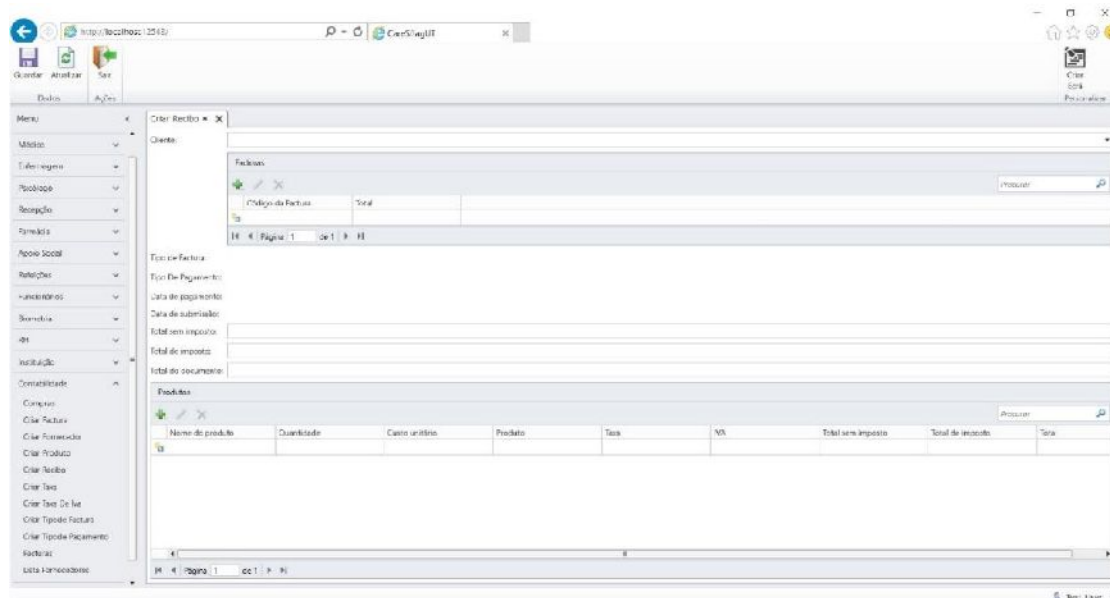


Figura 60 - criar recibo



O ecrã da figura 63 demonstra os recursos a que estão associados os clientes podendo desta forma verificar e alterar sempre que necessário os produtos existentes na instituição.

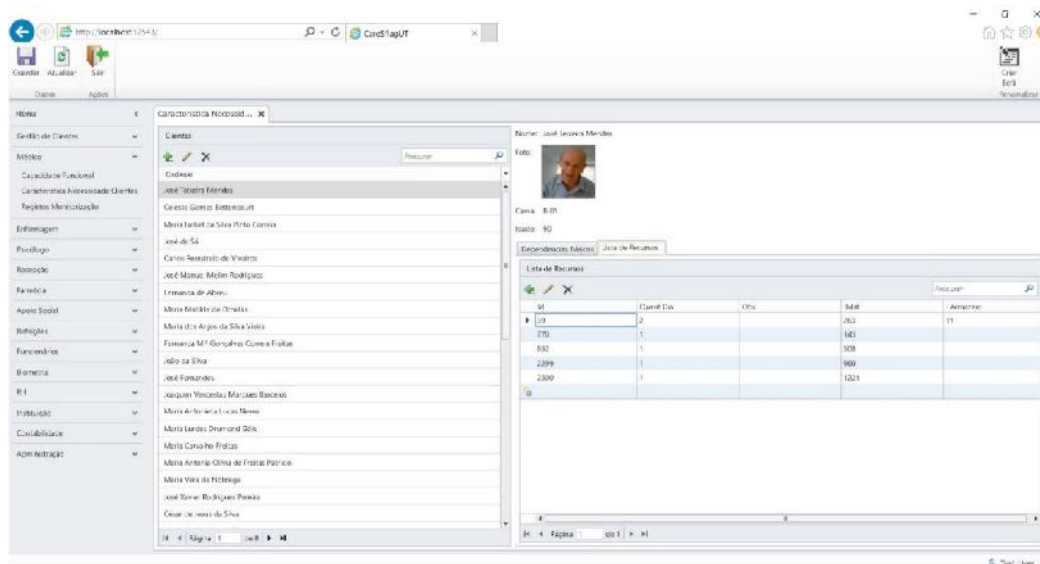


Figura 63 - lista de recursos por utente

A arquitetura de sistema atual é composta por uma instância de SQL 2008 r2, composta por três bases de dados:

- Caresflag (responsável pela gestão dos funcionários e dos stocks):
  - Aspnet\_Applications – nome das aplicações;
  - Aspnet\_Membership – Utilizadores da aplicação;
  - Aspnet\_Roles – perfis de utilizador;
  - Aspnet\_Users – relação entre o nome de utilizador e a encriptação do mesmo;
  - Aspnet\_UsersinRoles – Relação entre o utilizador e o perfil;
  - HR\_Bio\_TM\_PassingMode – Classificação de picagem no biométrico;
  - HR\_Bio\_TM\_VerifyMode – Tipo de picagem no biométrico;
  - Hr\_BioReg – registos do biométrico;
  - MM\_Armazens – Armazéns da instituição;
  - MM\_DocsStock – Responsável pelo report do Stock;
  - MM\_DocType – Tipo de documento;
  - MM\_Fornecedores – lista de fornecedores;
  - MM\_Materiais – lista de materiais;
  - MM\_MatGroup – grupo de materiais;
  - MM\_NiveisStock – quantificação dos níveis de stock;
  - MM\_Stock – stock geral;
  - MM\_StorageLocation – localização de cada armazenamento;
  - MM\_TbClientes – clientes da instituição;
  - MM\_TbContFatura – contagem e o ano da fatura;
  - MM\_TxIva – taxas de iva;
  - RolePermissions – autorização de ecrãs em relação ao perfil;
  - tlBioroteiro – Tempo de leitura dos biométricos;
  - tm\_FuncBiodevice – relação do número de funcionário com o id do biométrico;
  - tm\_BioDevice – Localização dos biométricos e respetivas características;
  - tm\_BioTemplates – informação dos funcionários no biométrico;
  - tmCategoriaProfissional – Categoria do funcionário;
  - tmDoc – documentos de cada funcionário;
  - tmEntidades – Entidades de contratação;
  - tmFuncionarios – lista de funcionários;
  - tmHorarios – descrição dos horários;
  - tmNiveisFuncionarios – ordenados dos funcionários;
  - tmObsFuncionarios – Observações dos funcionários;
  - tmStatusFuncionarios – Estados funcionários;
  - tmTipoPrestacaoServico – tipo de contrato;

- CaresflagMM (responsável pela faturação)
- MM\_ALLSTOCK – stock geral;
- MM\_Armazens – Armazéns da instituição;
- MM\_Clientes – clientes para facturar;
- MM\_Custos – Custo por cama;
- MM\_DocsStock – Responsável pelo report do Stock;
- MM\_DocType – Tipo de documento;
- MM\_FADOCTYPE – tipo de documento de faturação;
- MM\_FAHEADER – Cabeçalho da fatura;
- MM\_Falinhas – linhas das faturas;
- MM\_FAR\_FormaTerap – tipo de terapêutica;
- MM\_FAR\_PRIACTIVO – principio ativo;
- MM\_FAR\_TOMA – tipo de administração do medicamento;
- MM\_FARRecibos – Recibos;
- MM\_Fornecedores – lista de fornecedores;
- MM\_LinhasDoc – linhas de documento de material;
- MM\_Materiais – lista de materiais;
- MM\_MatGroup – grupo de materiais;
- MM\_Recursos – Lista de camas;
- MM\_StorageLocation – localização de cada armazenamento;
- MM\_TbCondPag – condições de pagamento;
- MM\_Txlva – taxas de iva;
- MM\_Unidades – unidades de medida;

## 8 Testes de Validação

Após a conclusão do desenvolvimento, segue-se a fase de testes. Nesta fase, os primeiros testes são internos ou também designados por unitários, que permitem validar separadamente as funcionalidades desenvolvidas e também filtrar erros básicos cometidos durante a programação.

Após todos os testes unitários estarem concluídos com sucesso, serão iniciados os testes integrados, ou seja, são efetuados testes em conjunto com o cliente, que testarão, não por funcionalidade mas pelo seu todo. Os testes mais relevantes foram desenhados, pelo utilizador e devem ter grande detalhe, tentando assim, cobrir o maior número de combinações possíveis, tanto para combinações válidas como inválidas, isto é, será necessário comprovar que o software devolve o resultado esperado, mas também é necessário perceber qual o seu comportamento aquando de introdução de informação inválida. Com este nível de detalhe, pretende-se garantir a solidez do projeto.

Podem ser identificadas funcionalidades imperfeitas que necessitem de correção, por isso o programador será o elemento de apoio presente nos testes. Qualquer dúvida na funcionalidade da aplicação será verificada e corrigida, caso isso se verifique necessário. Após correção, os utilizadores terão que voltar a testar a funcionalidade corrigida verificando se o problema foi efetivamente ultrapassado. A qualquer momento poderão ser inseridos casos de teste que os utilizadores achem pertinentes validar.

Após o trabalho de execução dos testes integrados estar finalizado com sucesso, será realizada a aceitação dos mesmos, ou seja, serão verificados os testes realizados avaliando se o resultado esperado se enquadra nos requisitos inicialmente identificados. Caso exista alguma dúvida, poderão ser solicitados novos testes ou correções pertinentes ao programador.

## 9 Conclusões

Nesta seção são apresentados os objetivos do trabalho que foram cumpridos, sugeridas algumas melhorias que podem ser realizadas no futuro e referidos os conhecimentos e competências adquiridos no desenvolvimento deste trabalho. A capacidade das ferramentas utilizadas excedeu as expectativas, pela possibilidade de desenvolver rapidamente uma aplicação com funcionalidades avançadas com tecnologias gratuitas.

### 9.1 Objetivos cumpridos

Foi feita uma pesquisa às aplicações existentes, tal como descrito no capítulo 5, com as características gerais da aplicação, não tendo sido descoberta nenhuma que encaixe completamente nos objetivos deste trabalho, embora haja algumas semelhanças.

Foram desenvolvidas algumas utilidades para responder às necessidades específicas que foram surgindo para realizar a interação, como o esconder do menu lateral esquerdo e superior, o reposicionamento de elementos, a janela de apresentação de conteúdos, o redimensionamento dos menus. Foi descrita a aplicação através de diagramas E-R, exibida a estrutura e tabelas da base de dados e descrito o desenvolvimento da aplicação desde a fase inicial até estar completamente funcional. Para assegurar a confidencialidade de certos dados, impedir o acesso indevido ou não permitido à aplicação ou partes dela foi realizada a implementação de medidas de segurança. A implementação das permissões dos utilizadores requereu a adaptação das partes do código da aplicação a apresentar a cada tipo de utilizador, que também é uma medida de segurança. A gestão de utentes, funcionários e utilizadores foi implementada e dá a possibilidade aos utilizadores com privilégios de administração de constituir as permissões para cada grupo de trabalho. Foi realizado o suporte da inserção, edição, consulta e eliminação de dados, como se trata de dados referentes à saúde os dados nunca são eliminados mas sim anulados. Uma funcionalidade vantajosa na eventualidade de eliminar dados sensíveis do utente, que serve para determinar um diagnóstico e um acompanhamento mais profissional e eficiente

Uma característica implementada e que pretende melhorar a coordenação é atualização em tempo real dos dados referentes ao apoio social na aplicação que representam os utentes em cada instituição diferente e os serviços utilizados por eles. Os restantes dados são igualmente atualizados em tempo real, mas a importância dos utilizadores e dos serviços utilizados é maior, por serem os principais dados na regularização que normalmente requer uma coordenação eficaz. A configuração das tabelas praticamente todas permite, entre outras informações, não limitar os dados apenas aos que já existem mas sim o próprio utilizador adicionar, alterar e eliminar sempre que for necessário.

Dos pontos enunciados na seção do Cronograma previsto, a etapa dos testes foi a única que não correspondeu ao previsto, já que não foram realizados testes que abrangessem mais cenários para além dos que estão descritos no capítulo das Funcionalidades implementadas. Nesse capítulo foram realizados testes ao funcionamento da aplicação e registrado o seu resultado. Mas, esse capítulo destinou-se apenas a demonstrar as funcionalidades e não o funcionamento de toda a aplicação. Assim, ficou por demonstrar o funcionamento de muitas partes da aplicação e o capítulo das “Funcionalidades implementadas” não pode ser considerado um capítulo de testes. Durante o desenvolvimento da aplicação foram experimentadas algumas dificuldades na implementação da Crystal Reports com o lightswitch, mas que foram ultrapassadas pela disponibilidade de informações na internet e pela orientação recebida.

## 10 Trabalho Futuro

A aplicação pode sempre ser melhorada, tanto para acrescentar funções úteis como para melhorar as que já existem. Ao longo do desenvolvimento foram surgindo ideias de melhorias, nomeadamente:

- A integração, não só com a segurança social, mas também com o serviço nacional de saúde;
- Integração no sistema integrado de rede de cuidados continuados;
- Introdução de serviços de relatórios para estatística global de utilização tanto por parte da segurança social como da instituição;

Ficam por realizar os testes em ambiente real para validar a aplicação, que não estava nos planos, mas o que é importante para verificar a aceitação a aplicação por parte das entidades a quem se destina e se necessário fazer as adaptações necessárias para responder às suas exigências.

Ficam assim algumas sugestões que não foram implementadas por não fazerem parte dos requisitos previstos inicialmente, embora os objetivos que se pretendiam foram cumpridos.

## 11 Referências

- Carta Social (2010), Rede de serviços e equipamentos, Relatório 2010, Gabinete de Estratégia e Planeamento, Ministério do Trabalho e da solidariedade social.
- Manual de processos-chave estrutura residencial para idosos, Instituto da Segurança Social Portuguesa
- Alves, André e Lopes, Arminda, "Do More with Less – A Framework for e-Government", 1st International Conference for Business Administration: "The economical and managerial 4 Contemporary Challenges and Future Prospects, Mutah University, Karak, Jordania, 2012
- Centro Social Amigos da Lardosa – Available at <http://csalardosa.pt/>
- Curso de Gestão de Instituições para Idosos (I) – Available at [www.socialgest.pt](http://www.socialgest.pt)
- Dick, B. (1997), Choosing action research [On line]. Available at [http://www.uq.net.au/action\\_research/arp/choice.html](http://www.uq.net.au/action_research/arp/choice.html)
- Kemmis, Stephen, and Robert McTaggart, ED. (1988) The Action Research Planner, Third Edition, Published by Deakin University
- O'Leary, Z. (2004) The Essential Guide to Doing Research, London, Sage
- Zachman, J. A (1987) Framework for Information Systems Architecture, IBM Systems Journal, 26 (3)
- [https://technet.microsoft.com/pt-pt/library/cc770472\(v=ws.10\).aspx](https://technet.microsoft.com/pt-pt/library/cc770472(v=ws.10).aspx)
- 
- As melhores empresas para trabalhar de 2011, Exame/Accenture
- 
- <http://www.egov4dev.org/success/definitions.shtml>
- O'Brien, Rory (1998), An Overview of the Methodological Approach of Action Research
-

- University of Manchester, eGovernment for Development Information Exchange
- 
- ISTVÁN NOVÁK, Beginning Microsoft Visual Studio LightSwitch Development
- 
- Jayaram Krishnaswamy, Microsoft Visual Studio LightSwitch Business Application Development